STATE VECTOR ESTIMATION IN THE PRESENCE
OF MEASUREMENT UNCERTAINTY

by
Leon K. Ekchian

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREES OF
BACHELOR OF SCIENCE
and
MASTER OF SCIENCE
at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
November, 1979

Signature redacted

Signature of Author
Signature redacted

Department of Electrical Engineering and Computer Science,
November 15, 1979

Signature redacted

Certified by
Thesis Supervisor

Signature redacted

Certified by
Company Supervisor

Signature redacted

Accepted by
Chairman, Departmental Committee on Graduate Students
DISCLAIMER OF QUALITY

Due to the condition of the original material, there are unavoidable flaws in this reproduction. We have made every effort possible to provide you with the best copy available. If you are dissatisfied with this product and find it unusable, please contact Document Services as soon as possible.

Thank you.

Due to the poor quality of the original document, there is some spotting or background shading in this document.
STATE VECTOR ESTIMATION IN THE PRESENCE OF MEASUREMENT UNCERTAINTY

by

Leon K. Ekchian

Submitted to the Department of Electrical Engineering and Computer Science on November 15, 1979 in partial fulfillment of the requirements for the Degrees of Master of Science and Bachelor of Science.

ABSTRACT

During the past decade considerable advances have been made in the general theory of adaptive state estimation and parameter identification. In this thesis the behavior of a specific class of stochastic systems (mobile satellite communications systems) is investigated. The major conclusions and results of this thesis are of two basic classes. The first class consists of specific results, which pertain directly to the Multipath Communication Case, while the second is comprised of their extensions to more general stochastic systems. Many mobile satellite communication systems exhibit specular multipath, which acts as distortion to the transmitted data. Generally, there is a singular specular return which can be characterized by a delay and reflection coefficient. Since the transmitted signal is received distorted with intersymbol interference, it is reasonable to view the independent bit source and channel with memory, as a corresponding finite state machine source and a memoryless channel. An algorithm is formulated which is called the Adaptive Joint Detection Estimation Algorithm (AJDEA). It generates a state estimate of the channel, i.e., the tap gain estimates of the finite state machine and an estimate of the unknown transmitted input data sequence. Actually, two alternative detectors are considered for generating the input data bit sequence estimate. The first is based on a maximum likelihood detection scheme, while the second is based on a weighted residual approach. Computer simulation results comparing the two alternatives are presented for not only the Multipath Communication Case, but also for two simple problems: the Scalar Case and the Two-Dimensional Vector Case. Insight gained from these two simpler problems proves to be very valuable in the analysis of the multipath case.

THESIS SUPERVISOR: Michael Athans

TITLE: Professor, Electrical Engineering and Computer Science, Director of LIDS
ACKNOWLEDGEMENTS

I greatly appreciate the sincere encouragement and guidance provided by Professor Michael Athans throughout my career at M.I.T., as both undergraduate counselor and graduate thesis supervisor. For this I will remain forever indebted. I would also like to express my appreciation to Dr. Terrence P. McGarty, who supervised the initial phase of this work at the Communications Satellite Corporation, Washington, D.C., and to Professor Al Drake for his support as graduate counselor.

Moreover, I would like to express my sincere thanks to various fellow graduate students, who provided frequent encouragement and the opportunity for detailed discussions, in particular Mr. Edward Y. Chow.

Finally, I would like to thank my parents for their unwavering support and encouragement during my years of study at M.I.T. Without them this thesis would never have been successfully completed.

This work was supported in part by the Office of Naval Research under Contract ONR/N00014-77-C-0532 at the Laboratory for Information and Decision Systems (M.I.T.) and the Communications Satellite Corporation under the Electrical Engineering Co-op Program.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABSTRACT</td>
<td>2</td>
</tr>
<tr>
<td>ACKNOWLEDGEMENTS</td>
<td>3</td>
</tr>
<tr>
<td>TABLE OF CONTENTS</td>
<td>4</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td>7</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>9</td>
</tr>
<tr>
<td>1. INTRODUCTION</td>
<td>10</td>
</tr>
<tr>
<td>1.1 Motivation</td>
<td>10</td>
</tr>
<tr>
<td>1.2 Background</td>
<td>20</td>
</tr>
<tr>
<td>1.3 Contributions of this Thesis</td>
<td>23</td>
</tr>
<tr>
<td>1.4 Overview of the Thesis Report</td>
<td>25</td>
</tr>
<tr>
<td>1.5 Notation</td>
<td>27</td>
</tr>
<tr>
<td>2. PROBLEM FORMATION</td>
<td>29</td>
</tr>
<tr>
<td>2.1 Introduction</td>
<td>29</td>
</tr>
<tr>
<td>2.2 Definition of the Three Specific Examples</td>
<td>31</td>
</tr>
<tr>
<td>3. DESCRIPTION OF THE ADAPTIVE JOINT DETECTION ESTIMATION ALGORITHM</td>
<td>34</td>
</tr>
<tr>
<td>3.1 Introduction</td>
<td>34</td>
</tr>
<tr>
<td>3.2 Properties of the AJDEA</td>
<td>41</td>
</tr>
<tr>
<td>3.3 Discussion</td>
<td>44</td>
</tr>
<tr>
<td>4. DISCUSSION OF SCALAR CASE</td>
<td>48</td>
</tr>
<tr>
<td>4.1 Introduction</td>
<td>48</td>
</tr>
<tr>
<td>4.2 Adaptation of the AJDEA - Alternative 1 for</td>
<td>4-</td>
</tr>
</tbody>
</table>
4.3 Adaptation of the AJDEA - Alternative 2 for the Scalar Case
4.4 Computer Simulation Results and Discussion
4.5 Summary

5. DISCUSSION OF THE TWO-DIMENSIONAL VECTOR CASE
5.1 Introduction
5.2 Adaptation of the AJDEA to the Two-Dimensional Vector Case
5.3 Computer Simulation Results
5.4 Special Cases with Step Changes of the State Vector
5.5 Further Computer Simulation Results
5.6 Summary

6. DISCUSSION OF THE MULTIPATH COMMUNICATION CASE
6.1 Introduction
6.2 Adaptation of the AJDEA to the Multipath Communication Case
6.3 Computer Simulation Results
6.4 Special Cases with Step Changes of the State Vector
6.5 Further Computer Simulation Results
6.6 Summary

7. COMMENTS AND CONCLUSIONS
7.1 Modifications
7.2 Summary of Conclusions 164
  7.2.1 Specific Conclusions 165
  7.2.2 General Conclusions 167
7.3 Suggestions for Future Research 168
  7.3.1 Specific 168
  7.3.2 General 171
APPENDIX A 172
APPENDIX B 184
APPENDIX C 205
APPENDIX D 210
REFERENCES 215
## LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Joint Detection Estimation Scheme</td>
<td>21</td>
</tr>
<tr>
<td>3.1</td>
<td>Application of the MMEA to the &quot;Limited&quot; General Case</td>
<td>36</td>
</tr>
<tr>
<td>3.2</td>
<td>Delimited Bank of Kalman Filters</td>
<td>40</td>
</tr>
<tr>
<td>4.1</td>
<td>$x = 10$</td>
<td>65</td>
</tr>
<tr>
<td>4.2</td>
<td>$x = 1.25$ - Alternative 1</td>
<td>67</td>
</tr>
<tr>
<td>4.3</td>
<td>$x = 1.25$ - Alternative 2</td>
<td>68</td>
</tr>
<tr>
<td>4.4</td>
<td>$x = -1.00$ - Alternative 1</td>
<td>70</td>
</tr>
<tr>
<td>4.5</td>
<td>$x = -1.00$ - Alternative 2</td>
<td>71</td>
</tr>
<tr>
<td>5.1</td>
<td>$x = [4.0 \ -5.0]^T$ - No Process Noise</td>
<td>87</td>
</tr>
<tr>
<td>5.2</td>
<td>$x = [4.0 \ -5.0]^T$ - Process Noise Added</td>
<td>88</td>
</tr>
<tr>
<td>5.3</td>
<td>$x = [4.0 \ -5.0]^T$ - Learning Sequence</td>
<td>91</td>
</tr>
<tr>
<td>5.4</td>
<td>$x = [2.0 \ -1.5]^T$ - Alternative 1</td>
<td>94</td>
</tr>
<tr>
<td>5.5</td>
<td>$x = [2.0 \ -1.5]^T$ - Alternative 2</td>
<td>95</td>
</tr>
<tr>
<td>5.6</td>
<td>$x = [-2.25 \ 2.25]^T$ - Alternative 1</td>
<td>97</td>
</tr>
<tr>
<td>5.7</td>
<td>$x = [-2.25 \ 2.25]^T$ - Alternative 2</td>
<td>98</td>
</tr>
<tr>
<td>5.8</td>
<td>$x = [-1.75 \ 1.75]^T$ - Alternative 1</td>
<td>100</td>
</tr>
<tr>
<td>5.9</td>
<td>$x = [-1.75 \ 1.75]^T$ - Alternative 2</td>
<td>102</td>
</tr>
<tr>
<td>5.10</td>
<td>Step Change from $x = [3.0 \ 8.0]^T$ to $[7.0 \ 5.0]^T$</td>
<td>103</td>
</tr>
<tr>
<td>5.11</td>
<td>Step Change from $x = [3.0 \ 8.0]^T$ to $[7.0 \ 5.0]^T$</td>
<td>111</td>
</tr>
<tr>
<td>5.12</td>
<td>Step Change from $x = [9.0 \ 4.0]^T$ to $[7.0 \ 5.0]^T$</td>
<td>112</td>
</tr>
<tr>
<td>5.13</td>
<td>Step Change from $x = [5.0 \ 3.0]^T$ to $[7.0 \ 5.0]^T$</td>
<td>114</td>
</tr>
</tbody>
</table>
5.14 Step Change from \( x = [4.0 \ 4.0]^T \) to \([7.0 \ 5.0]^T\)  
6.1(a), (b), (c) \( E = 14.0 \)  
6.2(a), (b), (c) \( E = 4.0 \)  
6.3(a), (b), (c) Alternative 1; \( E = 1.0 \)  
6.4(a), (b), (c) Alternative 2; \( E = 1.0 \)  
6.5(a), (b), (c) Step Change in the Reflection Coefficient from \( \rho = .9 \) to \( \rho = .5 \) at \( k = 25 \)  
A.1 Multiple Model Estimation Algorithm  
B.1 Delimited Bank of Kalman Filters  
B.2 Estimation Structure of AJDEA  
B.3 Probability Mass Function of the Data Bit  
Estimation Error \( \tilde{u}(k) \) Generated by Alternative 1
## LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1</td>
<td>Computer Simulation Results of the Scalar Case</td>
<td>64</td>
</tr>
<tr>
<td>5.1</td>
<td>Computer Simulation Results of the Two-Dimensional Vector Case</td>
<td>85</td>
</tr>
<tr>
<td>5.2</td>
<td>Step Jump from $\mathbf{x} = [4.0 \quad 4.0]^T$ to $[7.0 \quad 5.0]^T$</td>
<td>117</td>
</tr>
<tr>
<td>6.1</td>
<td>Computer Simulation Results of the Multipath Communication Case</td>
<td>131</td>
</tr>
</tbody>
</table>
CHAPTER 1

INTRODUCTION

1.1 Motivation

The motivation for this thesis originated from the communication problem studied by Schneider and McGarty in their paper "Reliable Satellite Communications for a Specular Multipath Channel" [29]. Specular multipath is present in many mobile satellite communication systems. It gives rise to distortions to the transmitted message sequence which can severely affect link performance. At present, there are two ways to cope with these distortions. The first is based on providing adequate link margin, while the second relies on the use of a large aperture antenna with adequate cutoff to attenuate the specular signal. Schneider and McGarty [29] propose a new technique that uses the multipath signal to actually enhance the performance of the overall system.

It is essential for the reader to be familiarized with this technique in order to follow the theoretical discussion and fully appreciate the results that are presented in this thesis report. Therefore, a detailed description of the model of the communication system and the processing technique is provided.

A source generates bits independently, one every T seconds, and transmits them over a channel which has a direct and specular
multipath portion. Let \( \{u(i)\} \) represent this transmitted bit sequence, where \( u(i) = \pm 1 \) with equal probability and \( i = 1, 2, 3, \ldots \). Each bit is modulated by \( \sqrt{E} \tilde{s}(t-iT) \) where \( \tilde{s}(t) \) is an orthonormal waveform and equal to zero outside of the interval \([0,T]\), while \( E \) is the signal energy. The bitwaveforms are further modulated before transmission by a set of spreading waveforms \( \tilde{\omega}(t) \). Hence the \( i \)th bit is modulated by \( \tilde{s}(t) \) where

\[
\tilde{s}(t) = \sum_{j=1}^{m} \tilde{\omega}\left(t-(j-1)\frac{T}{m}\right) \exp(j2\pi f_{0}t) \tilde{r}(t) \quad (1.1)
\]

\[
\tilde{\omega}\left(t-(j-1)\frac{T}{m}\right) = \begin{cases} 
\exp(j\phi_{j}) & \text{if } (j-1)\frac{T}{m}, \frac{j\pi}{m} \\
0 & \text{elsewhere} 
\end{cases} \quad (1.2)
\]

\( f_{0} \) is the carrier frequency

\[
\tilde{r}(t) = \frac{1}{\sqrt{T}} \quad (1.3)
\]

\( \phi_{j} = 0 \) or \( \pi \)

When this signal is transmitted over a channel from the satellite to a ground station, the received signal has been distorted by intersymbol interference as a result of the time delay experienced by the multipath return. This received signal is comprised of four terms: the direct signal \( \tilde{s}_{D}(t) \), the specular multipath return \( \tilde{s}_{S}(t) \), the diffuse multipath
return, and independent additive white Gaussian noise. The last two terms will be assumed to be white and combined into an equivalent front end noise, $n(t)$. Thus, the received signal can be expressed as

$$\tilde{r}(t) = \tilde{\mathcal{S}}_D(t) + \tilde{\mathcal{S}}_S(t) + n(t). \quad (1.4)$$

For most practical cases

$$\tilde{\mathcal{S}}_D(t) = \sum_{i=1}^{N} u(i)\tilde{s}(t-iT) \quad (1.5)$$

$$\tilde{\mathcal{S}}_S(t) = \sum_{i=1}^{N} u(i)\rho\tilde{s}(t-iT) \quad (1.6)$$

where

- $\rho$ is the complex reflection coefficient with a value between 0 and 1.
- $T$ is the excess multipath delay.
- $N$ is the transmitted message length.

and let

$$N_0 = \frac{2}{N}$$

be the spectral density height of $n(t)$.

Equation (1.4) can be rewritten as follows
\[ f(t) = \sum_{i=1}^{N} u(i)\tilde{h}(t-iT) + n(t) \quad (1.7) \]

where \( \tilde{h}(t) = \sqrt{E} \tilde{s}^*(t) + \rho \sqrt{E} \tilde{s}^*(t-T) \quad (1.8) \)

\[ \tilde{s}^*(t) = \frac{1}{\sqrt{T}} \sum_{j=1}^{m} \tilde{w} \left( t-(j-1)\frac{T}{m} \right) \]

However, since the transmitted signal is received distorted with intersymbol interference, bit by bit processing is no longer optimal and one resorts to the technique Forney proposed in his work on intersymbol interference. This technique is based on a dualism that exists between "the actual case of independent bit source and a channel with memory and the corresponding finite state machine source and a memoryless channel." Using this isomorphism, the optimal strategy is to use the Viterbi algorithm [29]. (For a brief discussion of the Viterbi algorithm, see Appendix A.2)

Schneider and McGarty investigate two different processing techniques, the "Optimum Scheme" and the "Robust Scheme." The first is based upon the technique proposed by Forney. However, the "Optimum Scheme" relies upon direct knowledge of the two multipath parameters: reflection coefficient \( \rho \) and time delay \( t \), which unfortunately must be formulated based on the measurements. Clearly, the overall performance will undoubtedly be
sensitive to measurement errors. In this thesis report we shall be particularly concerned with the latter of the two processing techniques. The "Robust Scheme" only requires direct knowledge of the multipath delay $\tau$ and forbears the need for the knowledge of the reflection coefficient $\rho$. It takes into account the information about the transmitted bit contained in the direct return, and also the redundant information in the multipath return.

Therefore, the "Robust Scheme" is a suboptimal technique. However, it is demonstrated by Schneider and McGarty that, for a large range of values of $\rho$, the performance of the "Robust Scheme" is only slightly worse than that of the "Optimum Scheme."

The "Robust Scheme" is implemented as follows. The output of the transmission model is initially supplied to a filter matched to $\tilde{s}^*(t)$. (See Equation (1.8)). The output of the filter is sampled, generating two different sequences of samples $\{z_1(k):i=1,2;k=1,2,3,\ldots\}$. The first sequence $\{z_1(k)\}$ is synchronized with the arrivals of the direct return. It is generated by sampling the matched filter output $T$ seconds after the beginning of the received transmission and thereafter every $T$ seconds. Conversely, the second measurement sequence $\{z_2(k)\}$ is synchronized with the arrivals of the specular multipath returns. It is generated by sampling the matched filter output $\tau-(K-1)T$ seconds after the beginning of the
received signal and thereafter every T seconds.

Thus, the following pair of equations is obtained:

\[ z_1(k) = \sqrt{E[u(k) + \rho R_s(KT-\tau)u(k-K)] + \rho R_s(\tau-(K-1)T)u(k-K+1)] + n_1(k)} \]  \hspace{1cm} (1.9)

\[ z_2(k) = \sqrt{E[R_s(KT-\tau)u(k) + R_s(\tau-(K-1)T)u(k-l)] + \rho u(k-K)] + n_2(k)} \]  \hspace{1cm} (1.10)

where \( R_s(\cdot) \) is the autocorrelation of \( s^*(\cdot) \).

\[ K = 1 + \frac{T}{T} \]  \hspace{1cm} (1.11)

\[ E[n_1^2(k)] = \begin{pmatrix} N_0 \end{pmatrix} \]  \hspace{1cm} i = 1, 2  \hspace{1cm} (1.12)

\[ E[n_1(k)n_2(k)] = \begin{pmatrix} N_0 \end{pmatrix} R_s(T-\tau) \]  \hspace{1cm} (1.13)

\[ E[n_1(k)n_2(k+1)] = \begin{pmatrix} N_0 \end{pmatrix} R_s(\tau) \]  \hspace{1cm} (1.14)

The model can be put into state variable form.

**Measurement Equation**

\[ z(k) = \Phi(k)x(k) + v(k) \]  \hspace{1cm} (1.15)
where \( \mathbf{x}(k) = \begin{bmatrix} 1 \\ p \mathbf{R}_s(\tau-(K-1)T) \\ p \mathbf{R}_s(\tau-T) \\ \mathbf{R}_s(\tau-KT) \\ \mathbf{R}_s(\tau-(K-1)T) \\ 0 \end{bmatrix} \) ; \( C(k) = \mathbb{E}\left[ \begin{bmatrix} u(k) & u(k+1) & u(k-K) & 0 & 0 & 0 \\ 0 & 0 & 0 & u(k) & u(k-1) & u(k-K) \end{bmatrix} \right] \)

\[ \mathbf{z}(k) = \begin{bmatrix} z_1(k) \\ z_2(k) \end{bmatrix} ; \quad \mathbf{v}(k) = \begin{bmatrix} n_1(k) \\ n_2(k) \end{bmatrix} \]

State Dynamics

\[ x(k+1) = x(k) \quad (1.18) \]

Note that the state is assumed to be constant over the duration of the information transmission.

The "Robust Scheme" furnishes an estimate of \( \{u(k)\} \), the data source output sequence, from the measurement vector sequence \( \{z(k)\} \) under the assumption that the state vector \( \mathbf{x}(k) \) is known. The procedure utilized in the "Robust Scheme" is a metric computer, which operates in a similar manner to the Viterbi algorithm. (See Section A.2). It utilizes the same routine as the Viterbi algorithm for extending, purging, and saving paths, and ultimately choosing a particular path through the trellis which corresponds to choosing an estimate \( \{u(k)\} \) of the input data sequence. A more detailed discussion
of the metric computer application to this problem can be found in Reference [29]. In particular, it is shown that if the noise sequences \( \{y(k)\} \) are completely uncorrelated, then the metric computer would actually be the Viterbi algorithm performing a maximum likelihood estimate of the input data sequence \( \{u(k)\} \).

However, the state vector \( x(k) \) is not typically known. Thus, before the Viterbi decoder can be employed, a channel state estimate must be generated. Schneider and McGarty propose using a Kalman filter to furnish \( \hat{x}(k/k) \), the estimate of \( x(k) \) given the measurement sequence \( \{z(k)\} \) and a known input sequence — a learning sequence. The reason for using a learning sequence of length \( L \) is that the transmission of these \( L \) bits provides complete knowledge of the matrix \( C(i) \) for \( i \in \{0, 1, 2, \ldots, L\} \). Hence, for \( i \) between \( 0 \) and \( L \), the complex nonlinear stochastic problem has been simplified to a linear multidimensional estimation problem. From standard filtering theory we know that, under these conditions, the Kalman filter provides the optimal (minimum mean square error) estimate of the state. Therefore, \( L \) bits are used to obtain \( \hat{x}(L/L) \), the state estimate of \( x(L) \) given the measurement sequence \( Z(L) = \{z(1), z(2), \ldots, z(L)\} \). Thereafter, during the actual data transmission length of \( N \) bits, the metric computer (Viterbi algorithm) is implemented by arbitrarily setting the unknown state vector \( x(k) \) to \( \hat{x}(L/L) \), the optimal
state vector estimate. The implicit assumption that has been made is that Equation (1.18) holds during the N bit transmission period, i.e., the state dynamics are invariant.

At this point, the following definitions are introduced, which will facilitate further discussion of the "Robust Scheme." The update interval is the time required to transmit the L bit learning sequence. The transmission interval is the time during which the actual N bit data sequence is transmitted. (Note that we have assumed that the processing times of the filters are negligible relative to the data bit transmission interval). The overall period is the time required to update the state estimate and transmit N actual data bits, i.e., the sum of the update interval and the transmission interval. The throughput of the system is the ratio of the number of actual data bits transmitted during the transmission interval to the total bits transmitted during the overall period, i.e.,

$$\text{Throughput} \triangleq \frac{N}{L + N} \quad (1.19)$$

Clearly, the L bit learning sequence constitutes an overhead cost. This cost is not exorbitant for those applications with relatively large values of N compared to L, i.e., one would ideally like to have the situation in which once the state vector estimate update cost has been incurred, the state dynamics will remain constant (i.e., Equation (1.18) is valid).
for a sufficiently long enough time to allow the transmission of a large number $N$ of actual data bits. Understandably, for situations in which state vector estimate updating is frequently necessary due to frequent state vector changes, the overhead costs associated with the "Robust Scheme" may be far too high.

Therefore, aside from the unavoidable update cost associated with the "Robust Scheme," one is also faced with the problem of determining a "reasonably safe" a priori estimate for the transmission interval, i.e., how long should one continue to transmit actual data bits before using a new learning sequence to update the state vector estimate.

The research performed for this thesis was devoted to studying these two problems in great detail and to devising an alternative scheme that generally provides a higher throughput (lower cost) and a higher level of adaptability to changes in the state vector $x(k)$. The extension does not encompass all stochastic changes of the state dynamics that can occur. Instead, only unknown step changes of $x(k)$ (with both unknown magnitude and time of occurrence) are considered. These unexpected step changes can occur frequently in the multipath communication case; for example, from unexpected step jumps of the reflection coefficient. We feel that this constitutes a step in the right direction and sets the trend for future work in handling more complex systems that depend
on unknown time-varying parameters.

Hence, a procedure is sought that simultaneously
detects the input binary data sequence and the state vector
with the highest possible throughput. Moreover, the algorithm
must constantly monitor the system so as to be able to track
any sudden changes of the channel state $x(k)$. A simple
graphical representation of the proposed joint detection-
estimation scheme is presented in Figure 1.1 for the multipath
communication case.

1.2 Background

The general area of stochastic estimation and identification
has recently received much attention. The reference section
at the end of this thesis report contains a lengthy list of
published papers. They have been chosen with the purpose
of maintaining a selection that encompasses a broad spectrum
of application. Examples of the application of the optimal
and suboptimal estimation techniques presented in these
papers are channel state vector estimation, multitarget
tracking, communication channel equalization and Bayesian
outlier rejection. Many more references can be found in the
IEEE Transactions on Automatic Control (Dec. 1974), an issue
specifically devoted to system identification.

Middleton and Esposito, in their paper "Simultaneous
Optimum Detection and Estimation of Signals in Noise" [32],
provide a good overview of the theory of joint detection and
\[ v(k) - x(k) \]

**ESTIMATOR**

\[ lw \]

**DETECTOR**

\[ S(x, C(u), u(k)) = c(k)x(k) \]

is the received signal.

\[ \hat{x}(k) \]

is the state vector estimate.

\[ \{\hat{u}(k)\} \]

is the estimate of the input binary sequence \{u(k)\}.

**Figure 1.1** JOINT DETECTION ESTIMATION SCHEME
estimation. They point out that optimal processing often requires the mutual coupling of detection and identification. (More about the significance of this mutual coupling will be presented in the derivation of the joint detection estimation algorithm later in this thesis report).

Bayesian techniques for estimation in dynamic systems, whose measurements are either the state plus noise or simply zero mean Gaussian noise, were formulated by Nahi [35] and Jaffer and Gupta [14]. These methods are related, but only conceptually, to the multipath channel problem. Thus, they will not be discussed in detail.

The basis for the formulation of the joint detection-estimation procedure proposed in this report is two well-known estimation and detection techniques. The first is an adaptive filtering and parameter detection scheme called the Multiple Model Estimation Algorithm, while the second is a maximum likelihood decoding scheme known as the Viterbi algorithm. A brief review of both procedures is provided in Appendix A. The Multiple Model Estimation Algorithm was originally introduced by Magill and later revised by Lainiotis [20, 21, 22] and Athans [4].

Bar Shalom and Tse [5] address a problem in multitarget tracking that is similar to the Multipath Communication Case. It has been recognized in target tracking, that there is often uncertainty associated with the origin of the received
measurements. This situation is prevalent in radar surveillance systems operating in dense multitarget or high clutter environments. The algorithm they propose incorporates all validated measurements in the track update by assigning to each measurement the a posteriori probability that it originated from the true target in track. These probabilities are used in formulating a computationally efficient track update equation. This procedure generates estimates that account for measurement origin uncertainties. (It is important to note that this paper was not known to the author until after the research for this thesis was completed).

1.3 Contributions of this Thesis

This thesis presents the results from a study of joint detection and estimation problems that are encountered in (time-varying) parameter dependent discrete systems. These results were obtained from a detailed analysis of three specific cases that are defined in Chapter 2. However, this limited study brings forth intrinsic properties that are common to more general situations. Thus, of importance, are the general conclusions made when the specific conclusions are extended to more general systems. These general conclusions are discussed in Chapter 7.

Listed below are the major contributions of this thesis.
1 - A major increase in the throughput of the multi-path communication system is obtained by formulating a joint detection-estimation algorithm that forbears the need for a costly learning sequence.

2 - The algorithm provides a means for constantly monitoring the dynamical system. Sudden step changes of the state vector are detected and tracked.

3 - The AJDEA consists of a coupled detector-estimator pair. Two alternative detectors are considered. Alternative 1 provides the data bit estimate based on a maximum likelihood detection scheme. Alternative 2 generates the data bit estimate by implementing a new weighted residual scheme. The estimator is a modified Kalman filter that accounts for the unknown observation matrix and uses artificial Gaussian noise to enhance the overall performance of the algorithm. The choice of the covariance of the artificial noise is based on subjective engineering judgement.

4 - The specific results of this thesis can be used to predict the properties of more complex, time-varying parameter dependent, discrete systems. In particular, the weighted residual approach appears extremely promising for application in such systems.
1.4 Overview of the Thesis Report

The major objective of this thesis is to formulate an efficient, high-throughput detection estimation algorithm for the Multipath Communication Case discussed in Section 1.1. However, we do not proceed directly to the derivation of the algorithm for the full-blown Multipath Communication Case. Instead, the algorithm is initially formulated for two simple cases: the Scalar Case and the Two-dimensional Vector Case. Insight gained from the indepth study of these two cases will prove to be valuable for the Multipath Communication Case.

Chapters 3, 4, 5, and 6 constitute the main body of this report. In Chapter 3, the algorithm is derived for the "limited" general case that encompasses the three specific cases of interest in this report. This is done so as to avoid the unnecessary redundancy of rederiving the algorithm for each specific case.

The ensuing three chapters correspond respectively to the three specific cases. In each of these chapters, the algorithm is adapted for the particular case at hand. Then an analytic discussion is presented, supported by a number of illustrative computer simulation results. The reason for choosing the above format is to make each of the three chapters self-contained and available for quick reference.

Listed below is a brief overview of each of the remaining chapters.
Chapter 2

The problem definition of the three specific cases is presented.

Chapter 3

The major properties of the detection-estimation algorithm for the "limited" general case is presented. However, the detailed derivations are deferred to Appendix B. The name chosen for the algorithm is the Adaptive Joint Detection Estimation Adaptation (AJDEA).

Chapter 4

Both analytic and computer simulation results from the basic Scalar Case implementation of the AJDEA are analyzed. Once again the detailed derivations are postponed to an appendix - Appendix C.

Chapter 5

The results of the application of the AJDEA to the Two-dimensional Case are presented. Moreover, conclusions reached in Chapter 4 are extended to vector cases. Finally, the adaptability of the AJDEA to sudden unexpected step changes of the state vector is investigated with on-line computer simulations.

Chapter 6

The AJDEA is applied to the most interesting of the three examples - the multipath communication case. Computer simulation results are presented and discussed.
The robustness of the algorithm is tested by introducing sudden step changes of the state vector. This is effectuated by changing the value of the reflection coefficient. Moreover, computer simulation results are provided.

Chapter 7

A modification to the state error covariance update equation of Alternative 2 is formulated. The major contributions of this thesis are summarized. Finally, suggestions for related future research topics are included.

Appendix A: Review of MMEA and Viterbi algorithm
Appendix B: Derivation of AJDEA
Appendix C: Analytic performance bounds of AJDEA - Alternative 1 for the Scalar Case.
Appendix D: Derivation of Error Covariance Modification

1.5 Notation

Listed below is the notation employed in this thesis. Matrices are denoted by upper case underlined letters: \( A, B, \) etc.

Vectors are denoted by lower case underlined letters \( x, y, z, \) etc.

 Scalars are denoted by lower case letters which are not underlined.

\[ |c| \] - magnitude of scalar \( c \)
\( M^T \) - denotes the transpose of \( M \)
\( M^{-1} \) - denotes the inverse of \( M \) (\( M \) is a square matrix)
\( k \) - scalar discrete time index
\( C(k) \) - observation matrix at time \( k \)
\( u(k) \) - data bit transmitted at time \( k \)
\( \hat{u}(k) \) - data bit estimate
\( x(k) \) - state vector at time \( k \)
\( x_i(k) \) - the \( i \)th element of the state vector at time \( k \)
\( \hat{x}(k/k) \) - state vector estimate given all the past measurements
\( \Sigma \) - error covariance matrix
\( \sum_{i=1}^{L} \) - summation from \( i=1 \) to \( L \)
CHAPTER 2

PROBLEM FORMULATION

2.1 Introduction

The three specific cases to be studied in this report are characterized by the following "limited" general case. The reason for adding "limited" is that the simplest general case is used in the algorithm derivation so as to minimize the complexity.

The "limited" general case is a discrete stochastic system, whose observation matrix depends on a time-varying parameter $u(k)$ that is either $\pm 1$. The discrete system is given by

**State Dynamics**

$$x(k+1) = x(k) + w(k)$$  \hspace{1cm} (2.1)

**Measurement Equation**

$$z(k) = C[u(k)]x(k) + v(k)$$  \hspace{1cm} (2.2)

Description of variables and coefficients:
- $k$ is the scalar discrete time index
- $x(k)$ is a time-varying parameter. Let the parameter value space be represented by $S$, which contains two possible scalar parameter values.
(S = {-1, 1}) with equal a priori probabilities.

- plant disturbance $w(k)$ is a $p$-dimensional vector that is a discrete white Gaussian noise sequence defined by the following statistics:

$$E\{w(k)\} = 0 \quad \text{for all } k \quad (2.3)$$

$$E\{w(k)w^T(j)\} = Q(k)\delta(k,j) \quad (2.4)$$

where $Q(k)$ is a $(p \times p)$ covariance matrix (2.5)

$$Q(k) = Q^T(k) \geq 0$$

$$\delta(k,j) = \begin{cases} 1 & k=j \\ 0 & k \neq j \end{cases} \quad (2.6)$$

- measurement vector $z(k)$ is an $m$-dimensional vector.

- measurement noise $v(k)$ is an $m$-dimensional vector that is a discrete white Gaussian noise sequence defined by the following statistics.

$$E\{v(k)\} = 0 \quad \text{for all } k \quad (2.7)$$

$$E\{v(k)v^T(j)\} = R(k)\delta(k,j) \quad (2.8)$$
$R(k)$ is a (mxm) covariance matrix  

$R(k) = R^T(k) > 0$

- moreover, $v(k)$ and $w(j)$ are independent for all values of $k$ and $j$, i.e.,

$$\text{cov}(w(k); v(j)) = 0 \quad \text{for all } k,j$$  (2.10)

- The observation matrix $C[u(k)]$ is (rxr) and is unknown, since it is dependent on an unknown parameter. Hereafter, we will use $C(k)$ instead of the cumbersome notation $C[u(k)]$ and simply remember that it is unknown.

The stochastic system defined above is governed by two random variables $C(k)$ and $x(k)$. An algorithm is sought that recursively identifies the time-varying observation matrix $C(k)$ and estimates the state vector $x(k)$.

2.2 Definition of the Three Specific Cases

The Adaptive Joint Detection Estimation Algorithm (AJDEA) was formulated for the following three specific cases:

I Scalar Case

The Scalar Case system is given by the following pair of discrete equations:

$$x(k+1) = x(k)$$  (2.11)
\[ z(k) = C(k)x(k) + v(k) \] (2.12)

where \( C(k) = u(k) \) (2.13)

\[
\begin{aligned}
& u(k) = \\
& \begin{cases} 
+1 & P[u(k) = 1] = \frac{1}{2} \\
-1 & P[u(k) = -1] = \frac{1}{2}
\end{cases}
\end{aligned}
\] (2.14)

For further definition of notation see Section 2.1.

### II Two-dimensional Vector Case

The Two-dimensional Vector Case system is given by the following pair of discrete equations:

\[ x(k+1) = x(k) \] (2.15)

\[ z(k) = \mathbf{C}(k)x(k) + v(k) \] (2.16)

where \( \mathbf{x}(k) = [x_1(k) \ x_2(k)]^T \) (2.17)

\( \mathbf{C}(k) = [u(k) \ u(k-1)] \) (2.18)

\[
\begin{aligned}
& u(k) = \\
& \begin{cases} 
+1 & P[u(k) = 1] = \frac{1}{2} \\
-1 & P[u(k) = -1] = \frac{1}{2}
\end{cases}
\end{aligned}
\] (2.19)

For further definition of notation see Section 2.1.

### III Multipath Communication Case

The Multipath Communication Case system is given by the following pair of discrete equations:
\[ x(k+1) = x(k) \] 

(2.20)

\[ z(k) = C(k)x(k) + v(k) \] 

(2.21)

For the definition of notation see Sections 1.1 and 2.1.

From Equation (1.16), (1.17)

\[
x(k) = \begin{bmatrix}
x_1(k) \\
x_2(k) \\
x_3(k) \\
x_4(k) \\
x_5(k) \\
x_6(k)
\end{bmatrix}
\]

(2.22)

For \( K = 4 \) (Refer to Equation (1.11) for definition of \( K \)).

\[
C(k) = \sqrt{E} \begin{bmatrix}
u(k) & u(k-3) & u(k-4) & 0 & 0 & 0 \\
0 & 0 & 0 & u(k) & u(k-1) & u(k-4)
\end{bmatrix}
\]

(2.23)

\[
\begin{bmatrix}
z_1(k) \\
z_2(k)
\end{bmatrix}; \quad v(k) = \begin{bmatrix}
n_1(k) \\
n_2(k)
\end{bmatrix}
\]
CHAPTER 3

DESCRIPTION OF THE ADAPTIVE JOINT
DETECTION ESTIMATION ALGORITHM

3.1 Introduction

It was emphasized in Chapters 1 and 2 that the research performed for this thesis was primarily directed at formulating a computationally efficient algorithm for recursively extracting the "history" of an unknown time-varying parameter \( u(k) \) and estimating the quasi-static unknown state vector \( x(k) \). The algorithm formulated and tested in this thesis is called the Adaptive Joint Detection Estimation Algorithm (AJDEA) for apparent reasons.

However, in Chapter 1 it was pointed out that actually two distinct alternatives of the AJDEA are considered. The estimator of the two alternatives of the AJDEA are, however, identical. The difference lies in the detectors of the two alternatives. This subtle difference in the detection of the unknown data bits constitutes a fundamental issue that will carefully be studied throughout this thesis.

The first alternative (Alternative 1) is a maximum likelihood detection scheme, while Alternative 2 is an algorithm that uses the a posteriori probabilities to generate a weighted estimate. In this chapter the major
properties of the AJDEA - Alternatives 1 and 2 are discussed. For the detailed derivation of these results the reader is referred to Appendix B.

The basis upon which the AJDEA is constructed is the Multiple Model Estimation Algorithm (MMEA) that is reviewed in Appendix A.1. At this point the reader is urged to review Appendix A.1 in order to familiarize himself with definitions and notation inherent to the MMEA.

The dynamical system that is used in the formulation of the AJDEA is the "limited" general case that was defined in Chapter 2 by the following pair of equations:

\[
x(k+1) = x(k) + w(k) \quad (3.1)
\]

\[
z(k) = C(k)x(k) + v(k) \quad (3.2)
\]

where the observation matrix \(C(k)\) is dependent on an unknown time-varying parameter \(u(k)\) that is either +1 at each time increment with equal a priori probability.

We are now ready to examine the problems that are encountered when one applies the standard MMEA to the "limited" general case for generating the state vector estimate \(\hat{x}(k/k)\). The purpose is to motivate the need for formulating a new computationally efficient algorithm.

Examining Figure 3.1 one sees that at \(k = 1\) the bank
Figure 3.1 APPLICATION OF THE MMEA TO THE "LIMITED" GENERAL CASE
of filters consists of two Kalman filters matched respectively to the two models (hypotheses) \( H_0 \) and \( H_1 \), which correspond to \( u(1) = 1 \) and \( u(1) = -1 \). The two filters in turn generate the state estimates \( \hat{x}_1(1/1) \) and \( \hat{x}_0(1/1) \). From Equation (A.27)

\[
\hat{x}(1/1) = P_1(1)\hat{x}_1(1/1) + P_0(1)\hat{x}_0(1/1) \tag{3.3}
\]

where \( P_1(1) \) and \( P_0(1) \) are weighting coefficients given by

\[
P_i(k) = \frac{P(\mathbf{z}(k) / H_i, \mathbf{Z}(k-1))}{\sum_{j=0}^{1} p(\mathbf{z}(k) / H_j, \mathbf{Z}(k-1))} \tag{3.4}
\]

where \( p(\mathbf{z}(k) / H_i, \mathbf{Z}(k-1)) \) is a Gaussian density defined in Equation (3.7) and

\[
\mathbf{Z}(k) = \{\mathbf{z}(1), \mathbf{z}(2), \ldots, \mathbf{z}(k)\}.
\]

The estimate \( \hat{x}(1/1) \) is the optimal minimum mean square error estimate of the state at \( k = 1 \).

Still continuing to examine Figure 3.1, note that at \( k = 2 \) the number of required Kalman filters has doubled. Each of the filters is matched to one of four possible models (hypotheses) \( H_{11}, H_{10}, H_{01} \) and \( H_{00} \), which correspond respectively to \( [u(1) = 1; u(2) = 1], [u(1) = 1; u(2) = -1] \),
\[ u(1) = -1; \ u(2) = 1 \] and \[ u(1) = -1; \ u(2) = -1 \]. The four filters in turn generate respectively, the following state estimates: \( \hat{x}_{11}(2/2), \hat{x}_{10}(2/2), \hat{x}_{01}(2/2), \) and \( \hat{x}_{00}(2/2) \). Moreover, the overall state estimate \( \hat{x}(2/2) \) is given by

\[
\hat{x}(2/2) = P_{11}(2)\hat{x}_{11}(2/2) + P_{10}(2)\hat{x}_{10}(2/2) + P_{01}(2)\hat{x}_{01}(2/2) + P_{00}(2)\hat{x}_{00}(2/2)
\]

(3.5)

i.e., the weighted sum of the elemental state estimates.

The weighting coefficients \( P_{11}(2), P_{10}(2), P_{01}(2), \) and \( P_{00}(2) \) are given by

\[
P_i(k) = \frac{P(z(2)/H_i, z(1))}{\sum_j p(z(2)/H_j, z(1))}
\]

(3.6)

where \( i \in \{11, 10, 01, 00\} \)

\( j \in \{11, 10, 01, 00\} \)

However, it is important to note that the state estimate \( \hat{x}(2/2) \) is still optimal in a minimum mean square error sense since the Multiple Model Estimation Algorithm has not yet been modified. But it is evident that there is a large computational burden associated with obtaining the optimal estimate since the number of required parallel Kalman filters increases twofold after each time increment, i.e., the bank of filters
grows exponentially as \(2^k\). This sequential exponential growth of the bank of Kalman filters is not only characteristic of the "limited" general case, but is encountered in many time-varying parameter dependent systems.

Clearly, the sequential growth of the bank of filters must be curtailed. The procedure chosen is to delimit the bank to two parallel Kalman filters. This procedure is illustrated in Figure 3.2. Note that the weighted average \(\hat{x}(1/l)\) is carried forward to \(k = 2\), instead of the individual filter outputs \(\hat{x}_1(1/l)\) and \(\hat{x}_0(1/l)\). Once again, \(P_i(k)\) is the probability that the \(i^{th}\) model is the actual model. By only carrying over the weighted average, essentially a "decoupling" of the \(k^{th}\) from the \((k-1)^{st}\) iteration has been obtained. The reason for this is that there is no need to increase the number of hypotheses to account for all possible paths in the past. Based on the values of \(P_1(k-1)\) and \(P_0(k-1)\), a detection operation is performed to determine the value of \(u(k-1)\). Hence at time \(k\), since the only uncertainty lies in the value of parameter \(u(k)\), only two hypotheses (\(H_0\) and \(H_1\)) are required instead of four.

An important issue that has been raised in this introductory section is complexity vs. optimality. In delimiting the growth of the bank of Kalman filters optimality has been sacrificed in gaining a tractable and computationally more efficient algorithm. It shall be demonstrated in the remaining chapters that, although suboptimal, the AJDEA provides excellent
Figure 3.2 DELIMITED BANK OF KALMAN FILTERS
state vector estimation and parameter identification under reasonable signal to noise conditions.

The complete derivation of the Adaptive Joint Detection Estimation Algorithm is given in Appendix B. In the next section a summary of the major properties of the AJDEA are presented and discussed.

3.2 Properties of the AJDEA

Listed below are the major properties of the AJDEA for the "limited" general case defined in Section 2.1.

Property 1: The bank of filters has been delimited to two Kalman filters. At each time increment there are two models (hypotheses $H_0$ and $H_1$) of the system.

Property 2: $P_i(k)$ is the probability that the $i^{th}$ model matches the actual system and is defined by the following equation:

$$P_i(k) = \frac{p(z(k)/H_i, Z(k-1))}{\sum_{j=0}^{1} p(z(k)/H_j, Z(k-1))} \quad i \in \{0, 1\} \quad (3.7)$$

where $p(z(k)/H_i, Z(k-1))$ is a Gaussian density equal to

$$\exp\left(-\frac{1}{2}(z(k)-\bar{Z}_i(k/k-1))^{T}(C_i(k)\Sigma(k/k-1)C_i^{T}(k)+R(k))^{-1}(z(k)-\bar{Z}_i(k/k-1))\right) \cdot (2\pi)^{-m/2}[\det(C_i(k)\Sigma(k/k-1))^{1/2} \cdot C_i^{T}(k)+R(k)]^{-1/2} \quad (3.8)$$
where \( m = \text{dim}[z(k)] \) \hspace{1cm} (3.9)

\[ Z(k-1) = \{ z(1), z(2), \ldots, z(k-1) \} \]

Finally, the sum of all \( P_i \)'s for a given \( k \) is one, i.e.,

\[ \sum_{i=0}^{1} P_i(k) = 1 \] \hspace{1cm} (3.10)

**Property 3:** It is shown in Appendix B that there are two alternatives of the AJDEA. The difference between the alternatives lies in the detector and not the estimator. The two alternative detectors are discussed in Property 4.

For the "limited" general case, the estimator equations of the AJDEA are derived in Appendix B - Equations (B.36), (B.37), and (B.38). These equations are repeated below. Note in particular that \( \hat{C}(k) \) is the estimate of the observation matrix.

**Predict Cycle**

\[ \hat{x}(k/k-1) = \hat{x}(k-1/k-1) \] \hspace{1cm} (3.11)

\[ \Sigma(k/k-1) = \Sigma(k-1/k-1) + Q(k-1) \] \hspace{1cm} (3.12)
Update Cycle

$$\hat{x}(k/k) = \hat{x}(k/k-1) + G(k)[z(k) - \hat{C}(k)\hat{x}(k/k-1)]$$  \hspace{1cm} (3.13)

$$\Sigma(k/k) = [I - G(k)\hat{C}(k)]\Sigma(k/k-1)$$ \hspace{1cm} (3.14)

$$G(k) = \Sigma(k/k-1)\hat{C}^T(k)[\hat{C}(k)\Sigma(k/k-1)\hat{C}^T(k) + R(k)]^{-1}$$ \hspace{1cm} (3.15)

Property 4: Parameter Identification

In Appendix B two alternatives of the AJDEA were formulated. The difference lies in the procedure used for determining the data bit estimate \( u(k) \). In Property 1 it was pointed out that the bank of Kalman filters has been delimited to two filters, i.e., at each time increment there are two models (hypotheses \( H_0 \) and \( H_1 \)) of the system.

Alternative 1 is a maximum likelihood detection procedure. The model (hypothesis) with the larger a posteriori conditional probability density is selected, i.e., the bank of Kalman Filters is always delimited to simply one filter since only the "most likely" filter is retained. Therefore, Alternative 1 is defined by the following hypothesis test:

Alternative 1: Maximum Likelihood Detection of \( u(k) \)

choose \( H_1: \hat{u}(k) = 1 \)

$$p(z(k)/H_1, Z(k-1)) \geq p(z(k)/H_0, Z(k-1))$$ \hspace{1cm} (3.16)

\( H_0: \hat{u}(k) = -1 \)
where \( p(z(k)/H_1, Z(k-1)) \) is defined in Equation (3.8).

On the other hand, Alternative 2 entails the use of a weighted residual approach to generate the data bit estimate \( \hat{u}(k) \). It is shown in Appendix B that the detector of the AJDEA - Alternative 2 is given by

**Alternative 2: Weighted Residual Estimation of \( u(k) \)**

\[
\hat{u}(k) = P_1(k)[1] + P_0(k)[-1] = P_1(k) - P_0(k)
\]

(3.17)

However, note that the value of \( \hat{u}(k) \) using Alternative 2 is not exactly \( \pm 1 \), but a decimal number between \(-1\) and \(+1\). This is the value that is used for \( \hat{u}(k) \) in constructing the observation matrix estimate \( \hat{a}(k) \) required by the AJDEA Equations (3.9)-(3.11). But clearly, the estimate of the transmitted bit sequence \( \{\hat{u}(k)\} \) must be a sequence of bits, i.e., \( \hat{u}(k) \) must be either \( \pm 1 \). The problem can be easily resolved by setting the estimate of \( u(k) \) to \( 1 \) if the decimal number \( \hat{u}(k) \) is greater than \( 0 \) and vice versa.

### 3.3 Discussion

In this chapter a new computationally efficient algorithm has been described that simultaneously furnishes the state vector estimate and the input binary data sequence estimate. Only the major properties of the AJDEA have been presented with very little supporting discussion. The reader is urged
to carefully study Appendix B for the complete derivation of the algorithm.

Clearly, it is important to generate a state vector estimate \( \hat{x}(k/k) \) which is as "close" as possible to the optimal (minimum mean square error) estimate that would be generated by the complete bank of Kalman filters. The difference between the state estimate provided by the AJDEA and the optimal estimate depends on the identification error of the observation matrix \( \hat{C}(k) \). Let this error be denoted by \( \tilde{C}(k) \), i.e.,

\[
\tilde{C}(k) = C(k) - \hat{C}(k)
\]

(3.18)

If the signal to measurement noise ratio is high, then detection errors are highly unlikely, i.e., the AJDEA state estimate is generally the optimal MMSE estimate. Thus, the smaller the \( \tilde{C}(k) \), the better the state estimate will be. Let \( \Gamma(k) \) be the correlation of \( \tilde{C}(k) \), i.e.

\[
\Gamma(k) = E(\tilde{C}(k)\tilde{C}^T(k))
\]

(3.19)

and consider the following situations:

"Very" high signal to noise ratio (SNR)

The complex nonlinear problem has been reduced to the standard linear estimation situation. Since the SNR is "very" high, the observation matrix estimate error \( \tilde{C}(k) \) is
essentially \( Q \) and \( \Gamma(k) = \Gamma(0) = Q \).

Basically, the AJDEA operates as the standard Kalman filter and furnishes the optimal MMSE estimate of \( \hat{x}(k) \).

Since the disturbances and \( x(0) \) are Gaussian, then the optimal estimate \( \hat{x}(k/k) \) is also the conditional mean, i.e.,

\[
\hat{x}(k/k) = E[x(k)/Z(k)]
\]  

"High to Moderate" Signal to Noise Ratio

Under these conditions there is a small uncertainty in matrix \( \hat{\Gamma}(k) \) (i.e., \(||\Gamma(k)|| < ||R(k)||\)). The state estimate \( \hat{x}(k/k) \) and the binary sequence estimate \( \{u(k)\} \) generated by the AJDEA have low root-mean-square (RMS) error and small probability of error respectively. Computer simulation results for all three specific cases under these conditions are given in the next three chapters. For example, in the multipath channel case, these conditions exist when the SNR is higher than 10 db and the probability of error is less than 5%.

"Low" Signal to Noise Ratio

Under these conditions the AJDEA does not provide an "excellent" estimate of the binary input sequence, i.e., there is significant probability of error (e.g. 20%). However, the state vector estimation part of the AJDEA still performs adequately and generates a reasonably "low" RMS error state estimate \( \hat{x}(k/k) \).
Consider the situation when the AJDEA has been run for a sufficient number of iterations until the residuals have decreased to a satisfactory level. (See Appendix A.1 for discussion of residuals). Let $\gamma$ be a sufficient number of iterations and $\hat{x}(\gamma/\gamma)$ the resulting state estimate. Since the binary sequence estimate furnished by the AJDEA is not satisfactory, a further operation must be performed. The Viterbi algorithm (Appendix A.2) is implemented with the assumption that $x(k) = \hat{x}(\gamma/\gamma)$, to generate the maximum likelihood estimate of the binary sequence. This final operation is not necessary for high SNR conditions because the probability of error levels of the AJDEA are satisfactory. Computer simulation of the AJDEA under these conditions for all three specific cases are presented in Chapters 4, 5, and 6. For example, in the multipath channel case, these conditions exist when the SNR is between 4 db and 10 db. However, computer simulation of the Viterbi algorithm is not performed since its performance results are widely available in the literature.

We are now prepared to proceed with the discussion of the three special cases that were outlined in Section 2.2.
CHAPTER 4

DISCUSSION OF SCALAR CASE

4.1 Introduction

The Scalar Case is the first of the two preliminary cases that are investigated in this thesis report. It is felt that by initially studying the application of the Adaptive Joint Detection Estimation Algorithm to this simple case, insight will be gained that will prove to be invaluable in studying the vector cases in the ensuing two chapters.

In Section 4.2 the general vector equations of the AJDEA - Alternative 1 which were formulated in Chapter 3 and Appendix B, are adapted for the Scalar Case. Moreover, analytic performance results derived in Appendix C are discussed. In Section 4.3 the general vector equations of the AJDEA - Alternative 2 are adapted for the Scalar Case. Computer simulation results obtained from applying both versions of the AJDEA are presented and discussed in Section 4.4. Finally, a brief summary of the major issues elucidated in this chapter is given in Section 4.5.

The Scalar Case should be regarded as a contrived initial example used to understand the operation of the AJDEA before proceeding to the Two Dimensional Vector Case and finally the Multipath Communication Case. Actually, by studying the problem definition of the system given below, it is apparent that the
Scalar Case could be thought of as an extremely simplified version of the Multipath Communication Case, which was expounded in Chapter 1. In this case there is no multipath delay and the measurement is merely a function of the most recent transmitted bit corrupted with white Gaussian noise.

The system studied in this chapter is defined by the following pair of discrete state space equations:

**State Dynamics**

\[ x(k+1) = x(k) \]  \hspace{1cm} (4.1)

**Measurement Equation**

\[ z(k) = C(k)x(k) + v(k) \]  \hspace{1cm} (4.2)

where \( x(k) \) is a scalar state vector, \( z(k) \) is the scalar measurement vector. The noise source \( v(k) \) is assumed to be a zero mean white Gaussian noise of covariance \( R \). Note that the system has no process noise, i.e., the state \( x(k) \) is constant over time. The initial state \( \hat{x}(0/0) \) is assumed to be Gaussian with mean \( x_0 \) and covariance \( P(0/0) \).

The observation matrix \( C(k) \) is an unknown scalar given by

\[ C(k) = u(k) \]  \hspace{1cm} (4.3)

where \( u(k) \) is the transmitted data bit that is equal to 1 with equal a priori probability. Moreover, the input binary
data sequence \{u(k)\} is white, i.e., u(k) and u(j) are uncorrelated for k ≠ j.

The objective is to obtain simultaneously adaptive estimates of the input binary sequence \{u(k)\} and the state x(k) with the highest throughput possible. (For the definition of throughput see Equation (1.19)). The procedure that is used in attaining this objective is the AJDEA of Chapter 3.

4.2 Adaptation of the AJDEA - Alternative 1 for the Scalar Case

It has already been emphasized that this system is not driven by process noise. However, it will be shown that for proper operation of the algorithm artificial Gaussian process noise with zero mean and covariance matrix Q is added.

The vector equations of the AJDEA, (3.11) through (3.15), reduce to simple scalar equations and the observation matrix estimate \( \hat{\theta}(k) \) is simply the scalar data bit maximum likelihood estimate \( \hat{u}(k) \). Therefore, the algorithm is given by the following set of equations:

**Predict Cycle**

\[
\hat{x}(k/k-1) = \hat{x}(k-1/k-1) \quad (4.4)
\]

\[
\Sigma(k/k-1) = \Sigma(k-1/k-1) + Q \quad (4.5)
\]

**Update Cycle**

\[
\hat{x}(k/k) = \hat{x}(k/k-1) + G(k)[z(k) - G(k)\hat{x}(k/k-1)] \quad (4.6)
\]
\[ E(k/k) = [1 - G(k)\hat{G}(k)]E(k/k-1) \quad (4.7) \]

\[ G(k) = \frac{\Sigma(k/k-1)\hat{G}(k)}{\Sigma(k/k-1) + R} \quad (4.8) \]

We are now ready to proceed with an analytic study of the operation of the AJDEA for the Scalar Case. Now by substituting for \( G(k) \) in Equation (4.7) and using the fact that \( \hat{G}(k) = \pm 1 \) it can be shown that

\[ \Sigma(k/k) = \frac{R\Sigma(k/k-1)}{\Sigma(k/k-1) + R} \quad (4.9) \]

Substituting for \( \Sigma(k/k-1) \) in Equation (4.9) we obtain

\[ \Sigma(k/k) = \frac{R[\Sigma(k-1/k-1) + Q]}{[\Sigma(k-1/k-1) + Q] + R} \quad (4.10) \]

In order to simplify the notation let \( \Sigma(0/0) = \Sigma_0 \). Therefore, by using Equation (4.10) it can be shown that

\[ \Sigma(1/1) = \frac{R[\Sigma_0+Q]}{[\Sigma_0+Q] + R} \quad (4.11) \]

and

\[ \Sigma(2/2) = \frac{R[\Sigma_0+Q][1+Q] + R^2Q}{[\Sigma_0+Q][2R+Q] + RQ + R^2} \quad (4.12) \]
There is no loss of generality in setting $R = 1$. Moreover, since $Q$ represents additive artificial noise it can be omitted in this analytic formulation. Therefore, Equation (4.12) reduced to

$$
\Sigma(2/2) = \frac{\Sigma_0}{1+2\Sigma_0}
$$

(4.13)

By extrapolating to time $k$ we obtain

$$
\Sigma(k/k) = \frac{\Sigma_0}{1+\kappa\Sigma_0}
$$

(4.14)

The next step is the evaluation of Equation (4.8) for the gain $G(k)$. Hence

$$
G(k) = \frac{\Sigma(k-1/k-1)\hat{u}(k)}{\Sigma(k-1/k-1)+1}
$$

(4.15)

By comparing Equation (4.9) to (4.15) it is clear that

$$
G(k) = \hat{u}(k)\Sigma(k/k)
$$

(4.16)

Having evaluated $\Sigma(k/k)$ and $G(k)$ we are ready to solve for the state estimate $\hat{x}(k/k)$ with Equation (4.6). Thus

$$
\hat{x}(k/k) = \hat{x}(k/k-1)+\hat{u}(k)\Sigma(k/k)[z(k)-\hat{u}(k)\hat{x}(k/k-1)]
$$

(4.17)
Substituting for $\Sigma(k/k)$ in Equation (4.17) and realizing that $\hat{\Sigma}^2(k) = 1$, one obtains the following equation for $\hat{x}(k/k)$:

$$\hat{x}(k/k) = \frac{\sum_{i=1}^{k} \hat{\mu}(i)z(i)}{1 + k\Sigma_0}$$

(4.18)

The adaptation of the AJDEA Equations (3.13) through (3.15) to the Scalar Case is now complete. Equations (4.14), (4.16), and (4.18) completely describe the estimation part of the AJDEA. Note that for this simple case the equations have been expressed in closed form.

The next step is the adaptation of the detection part of the AJDEA to the Scalar Case. The same definition for the two hypotheses $H_0$ and $H_1$ will be used for the Scalar Case, i.e., $H_0$ is the hypothesis (model) that $(u(k) = -1)$, while $H_1$ is the hypothesis (model) that $(u(k) = 1)$.

The maximum likelihood hypothesis test that is the basis of Alternative 1 was defined by Equation (3.16), namely

choose $H_1 : \hat{\mu}(k) = 1$

$$p(z(k)/H_1, Z(k-1)) > p(z(k)/H_0, Z(k-1))$$

(4.19)

$H_0 : \hat{\mu}(k) = -1$
where \( p(z(k)/H_1, Z(k-1)) \) is a Gaussian density equal to

\[
\exp\left[-\frac{(z(k)-\hat{x}(k-1/k-1))^2}{2(\Sigma(k-1/k-1)+1)}\right]
\frac{1}{(2\pi)^{\frac{3}{2}}[\Sigma(k-1/k-1)+1]^\frac{3}{2}}
\]

(4.20)

and

\[
Z(k) = \{z(1), z(2), z(3), \ldots, z(k)\}
\]

(4.21)

With a few algebraic manipulations it can be shown that the maximum likelihood test of Equation (4.19) can be reduced to the following pair of simple tests

\[
\begin{align*}
\text{choose } H_1: \hat{u}(k) & = 1 \\
\text{for } x > 0 & \\
\quad z(k) & > 0 \\
H_0: \hat{u}(k) & = -1
\end{align*}
\]

which generates \( \{\hat{u}(k)\}_{\text{pos}} \)

(4.22)

\[
\begin{align*}
\text{choose } H_1: \hat{u}(k) & = 1 \\
\text{for } x < 0 & \\
\quad z(k) & < 0 \\
H_0: \hat{u}(k) & = -1
\end{align*}
\]

which generates \( \{\hat{u}(k)\}_{\text{neg}} \)
The reason for having expressed the binary hypothesis test of Equation (4.19) in the format of Equation (4.22) is to point out a crucial issue. Note that, depending on whether $x(k)$ is positive or negative, either $\{\hat{u}(k)\}_\text{pos}$ or $\{\hat{u}(k)\}_\text{neg}$ respectively is generated by the AJDEA. Also note that $\{\hat{u}(k)\}_\text{neg}$ is the complement of $\{\hat{u}(k)\}_\text{pos}$. Moreover, it can be seen from Equations (4.18) and (4.19) that the sign (positive or negative) of the state estimate is determined by the sign of $\hat{x}(0/0)$. It is clear that $\hat{x}(k/k)$ will retain for all $k$ the original sign assigned to it. Therefore, the state estimate must always be initialized to zero, i.e., $\hat{x}(0/0) = 0$, since at $k = 0$ there is no a priori information to induce one to bias $\hat{x}(0/0)$ in either the positive or negative direction. Consequently, the first state estimate is critical since if it has the wrong sign, the state estimation process would be moving in the wrong direction and tracking $-x$ instead of $x$. Moreover, the algorithm would be generating thereafter the complement of the data sequence estimate it should be generating.

Hence, it is important to ensure, with high reliability, that the sign of $\hat{x}(1/1)$ be correct. One technique is using an initialization bit $u(1)$, i.e., $u(1)$ is known at the receiver a priori and thus, no detection is performed at $k = 1$. Assume that it is agreed upon that $u(1)$ will always be 1.

Now from Equation (4.2) we know that

$$z(1) = u(1)x(1) + \nu(1)$$  \hspace{1cm} (4.23)
Since \( u(1) = 1 \), then in general \( x(1) \) should have the same sign as the measurement \( z(1) \). The only problem that can arise is if the instantaneous value of the corruptive measurement noise is "too" large, i.e., if \( v(1) \) is larger than \( x(1) \) and \( v(1) \) has the opposite sign from \( x(1) \). Under these very "low" signal to noise (SNR) conditions the measurement \( z(1) \) can therefore drive the AJDEA astray and thereafter have it converge on the "complementary" solution.

In order to safeguard against such catastrophic failures, voting can be performed at the end of the transmission. It will ensure with some nominal level of confidence that the sign of \( \hat{x}(k/k) \) is correct and that \( \{\hat{u}(k)\} \) is the "correct" binary sequence estimate. There are numerous voting techniques that can be employed, depending on the level of confidence desired and the maximum cost that can be sustained. Once again, the cost is loss of throughput level since known bits must be transmitted per overall (transmission) period in order to establish a voting procedure. For example, three known bits can be transmitted and a majority rule can be implemented to check if the "correct" solution has been obtained. If the test fails, then the final AJDEA output is the negative value of \( \hat{x}(k/k) \) and the complement of the data bit sequence estimated that was originally generated. This technique is a very simple option. A more sophisticated technique is the use of "redundancy check" bits that are
transmitted at the end of the transmission interval (defined in Chapter 1). These bits can be generated by the modulo-2 sum of various predetermined data bits known both at the transmitter and receiver. This offers the operator of the receiver the option of testing the received data sequence for detection errors.

In this thesis, we are not greatly interested in investigating voting schemes for error detection. We shall leave it to the receiver operator to determine which voting scheme, if any at all, is necessary. In fact, for most reasonable SNR levels the convergence of the AJDEA to the "complementary" solution is unlikely if an initialization bit \( u(1) \) is transmitted.

Therefore, we will only rely on the crucial initialization bit in the Scalar Case simulations. Thus, from Equation (1.19), the throughput is given by

\[
\text{Throughput} = \frac{\Delta N}{1 + N} \quad (4.24)
\]

where \( N \) is the length of the transmitted data sequence.

The last topic that will be discussed in this section is the analytic performance results obtained in Appendix C for the Scalar Case. It is shown that under reasonable signal to noise ratio conditions
Equation (4.25) essentially asserts the intuitive feeling we had, that for reasonable SNR conditions the algorithm would perform well and there would be a minimal number of detection errors. It was pointed out earlier that due to the initialization problems of the sign \( \hat{x}(k/k) \) one cannot guarantee that the estimate is unbiased in the traditional sense. However, it should be expected that as the SNR decreases, average bias errors arise and increase.

4.3 Adaptation of the AJDEA - Alternative 2 for the Scalar Case

The vector equations of the AJDEA, (3.11) through (3.15), reduce to simple scalar equations and the observation matrix estimate \( \hat{C}(k) \) is constructed from the weighted bit estimate \( \hat{Q}(k) \). Thus, for the Scalar Case, the AJDEA Equations (3.11) through (3.15) simplify to

\[
\hat{x}(k/k-1) = \hat{x}(k-1/k-1) \quad (4.27)
\]

\[
\Sigma(k/k-1) = \Sigma(k-1/k-1) + Q \quad (4.28)
\]
Update Cycle

\[ \hat{x}(k/k) = \hat{x}(k/k-1) + G(k)[z(k) - \hat{u}(k)\hat{x}(k/k-1)] \]  
(4.29)

\[ \Sigma(k/k) = [I - G(k)\hat{u}(k)]\Sigma(k/k-1) \]  
(4.30)

\[ G(k) = \frac{\Sigma(k/k-1)\hat{u}(k)}{\hat{u}^2(k)\Sigma(k/k-1) + R} \]  
(4.31)

We are now ready to proceed with an analytic study of the operation of the AJDEA - Alternative 2 for the Scalar Case. In Section 4.2 the computational effort was alleviated by realizing that there is no loss of generality in setting \( R = 1 \) and omitting the additive artificial noise \( (Q = 0) \) for the analytic formulation. (However, it shall be pointed out later in this chapter and in ensuing chapters, that artificial process noise is necessary for the proper operation of the algorithm and, therefore, will be used in computer simulations).

Moreover, the following assumptions for the mean and covariance of \( \hat{x}(0/0) \) are made.

\[ x_0 = 0 ; \Sigma(0/0) = \Sigma_0 \]  
(4.32)

Next, by substituting for \( G(k) \) in Equation (4.30) and using Equation (4.31), it can be shown that
\[
\Sigma(k/k) = \frac{\Sigma(k-1/k-1)}{\hat{u}^2(k)\Sigma(k-1/k-1) + 1}
\]  

(4.33)

Since the initial covariance \(\Sigma(0/0) = \Sigma_0\), it can be shown that

\[
\Sigma(k/k) = \frac{\Sigma_0}{1 + \Sigma_0 \sum_{i=1}^{k} \hat{u}^2(i)}
\]  

(4.34)

The next step is the evaluation of Equation (4.31) for the gain \(G(k)\). Thus

\[
G(k) = \frac{\hat{u}(k)\Sigma(k-1/k-1)}{\hat{u}^2(k)\Sigma(k-1/k-1) + 1}
\]  

(4.35)

Therefore

\[
G(k) = \hat{u}(k)\Sigma(k/k)
\]  

(4.36)

Having evaluated \(\Sigma(k/k)\) and \(G(k)\) we are now prepared to solve for the state estimate \(\hat{x}(k/k)\) with Equation (4.29). Thus

\[
\hat{x}(k/k) = \hat{x}(k/k-1) + \frac{\hat{u}(k)\Sigma(k-1/k-1)}{\hat{u}^2(k)\Sigma(k-1/k-1)} [z(k) - \hat{u}(k)\hat{x}(k/k-1)]
\]  

(4.37)
By substituting Equation (4.34) into (4.37) it can be shown that

\[ \hat{x}(k/k) = \frac{\sum_{i=1}^{k} \hat{u}(i)z(i)}{1 + \sum_{i=1}^{k} \hat{u}^2(i)} \]  

(4.38)

The adaptation of the Adaptive Joint Detection Estimation Algorithm Equations (3.13) through (3.15) to the Scalar Case is now complete. Equations (4.34), (4.36), and (4.38) completely describe the estimation part of the AJDEA - Alternative 2. Moreover, note that the equations have been expressed in closed form.

The next step is the adaptation of the detection part of the AJDEA to the Scalar Case. Once again, the same definition for the two hypotheses \( H_0 \) and \( H_1 \) will be used for the Scalar Case, i.e., \( H_0 \) is the hypothesis (model) that \( u(k) = -1 \), while \( H_1 \) is the hypothesis (model) that \( u(k) = 1 \).

The estimate \( \hat{u}(k) \) is generated under Alternative 2 by the following weighted sum

\[ \hat{u}(k) = P_1(k)[1] + P_0(k)[-1] = P_1(k) - P_0(k) \]  

(4.39)

where \( P_i(k) (i \in \{0,1\}) \) is the a posteriori probability, defined by Equation (3.7), that the \( i \)th model matches the
actual system. For the Scalar Case, Equation (3.7) reduces to:

\[
P_1(k) = \frac{P(z(k)/H_1, Z(k-1))}{\sum_{j=0}^{1} p(z(k)/H_j, Z(k-1))}; i = \{0, 1\} \tag{4.40}
\]

where \( p(z(k)/H_i, Z(k-1)) \) is a Gaussian density given by

\[
\exp\left[-\left(z(k)-\hat{x}(k-1/k-1)\right)^2/2\Sigma(k-1/k-1) + 1\right]/(2\pi)^{\frac{1}{2}}[\Sigma(k-1/k-1) + 1]^{\frac{1}{2}} \tag{4.41}
\]

and

\[
Z(k) = \{z(1), z(2), \ldots, z(k)\} \tag{4.42}
\]

Once again, as in the case of Alternative 1, the algorithm requires an initialization bit \( u(1) \). (See Section 4.1). Thus, the throughput of the system is once again given by

\[
\text{Throughput} = \frac{N}{1 + N}
\]

where \( N \) is the length of the transmitted data sequence.

The adaptation of both the estimation and detection operations of the AJDEA to the Scalar Case is now complete. We are ready to proceed to Section 4.4 for the presentation of computer simulation results for both Alternatives 1 and 2.
and a general discussion.

4.4 Computer Simulation Results and Discussion

The next step is to proceed with the discussion of the computer simulation results under both "high to moderate" and "low" signal to noise ratios.

The measurement sequence \( \{z(k)\} \) is generated by first using a random number generator with different seeds to generate the transmitted data sequence \( \{u(k)\} \) and a Gaussian distributed noise sequence \( \{v(k)\} \) with zero mean and unit variance. Then by choosing a value for the state \( x \) and implementing Equation (4.2) the measurement sequence \( \{z(k)\} \) is obtained.

In Table 4.1 all the simulation results that are discussed in this section are listed for quick reference with their associated figure numbers.

"High to Moderate Signal to Noise Ratio (SNR)"

Under "high to moderate" SNR conditions the transmitted data bit estimate \( \hat{u}(k) \) generated by Alternative 1 and Alternative 2 are essentially equal since the a posteriori probabilities \( P_1(k) \) and \( P_0(k) \) are very close to 1 and 0 or vice versa.

Figure 4.1 is the presentation of the results obtained using Alternative 1 for the case in which the state \( x = 10 \). After fifty iterations the state estimate \( \hat{x}(k/k) = 9.75 \) with an RMS error over fifty iterations of \(.48\). The unknown data bit sequence estimate \( \{\hat{u}(k)\} \) is excellent with no detection.
<table>
<thead>
<tr>
<th>DESCRIPTION</th>
<th>$\hat{x}(50/50)$</th>
<th>RMS ERROR OF $\hat{x}(k/k)$</th>
<th>NUMBER OF DETECTION ERRORS</th>
<th>FIGURES</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x = 10.0$</td>
<td>9.75</td>
<td>1.3</td>
<td>0</td>
<td>4.1</td>
</tr>
<tr>
<td>artificial noise $Q = .01$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$x = 1.25$</td>
<td>1.71</td>
<td>.48</td>
<td>5</td>
<td>4.2</td>
</tr>
<tr>
<td>artificial noise $Q = .01$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Alternative 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$x = 1.25$</td>
<td>1.47</td>
<td>.25</td>
<td>5</td>
<td>4.3</td>
</tr>
<tr>
<td>alternative noise $Q = .01$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Alternative 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$x = 1.00$</td>
<td>-1.19</td>
<td>.18</td>
<td>10</td>
<td>4.4</td>
</tr>
<tr>
<td>artificial noise $Q = .01$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Alternative 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$x = -1.00$</td>
<td>-1.45</td>
<td>.32</td>
<td>10</td>
<td>4.5</td>
</tr>
<tr>
<td>artificial noise $Q = .01$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Alternative 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**TABLE 4.1**

COMPUTER SIMULATION RESULTS OF THE SCALAR CASE
Fig. 4.1 $x = 10$. 
errors made out of forty-nine transmitted bits. Numerous other "high to moderate" SNR cases were simulated with similar excellent results. These results are not presented so as to avoid redundancy. Based on these results we can resolutely assert that the Adaptive Joint Detection Estimation Algorithm performs excellently under "high to moderate" SNR conditions. The next step is to investigate the performance of the AJDEA under lower SNR conditions.

"Low" Signal to Noise Ratio

As one would expect intuitively the Adaptive Joint Detection Estimation Algorithm does not generally perform as well under "low" SNR conditions as it did under "high to moderate" SNR conditions. The number of detection errors and the estimation errors can generally be expected to increase. In this section computer simulation results implementing both Alternatives 1 and 2 are presented and compared.

Figure 4.2 is the presentation of the results obtained using Alternative 1 for the case in which the state $x = 1.25$. After fifty iterations the state estimate $\hat{x}(k/k)$ is 1.71 with a RMS error of .43. The number of detection errors in the data bit sequence estimate is five. As expected, the number of detection errors have increased.

On the other hand, Figure 4.3 is the presentation of the results obtained using Alternative 2 for the same case
Fig. 4.2  $x = 1.25$

Alternative 1
as in Figure 4.2, i.e., the same input sequence \{u(k)\} and measurement sequence \{z(k)\} are used and the state \(x\) is 1.25. After fifty iterations the state estimate is 1.47. By comparing the RMS errors of the two Alternatives (Figures 4.2 and 4.3), one concludes that Alternative 1 (maximum likelihood detection) generates a better state estimate than Alternative 2 (weighted residual estimation).

The next case that is considered is depicted in Figures 4.4 and 4.5. Figure 4.4 is the representation of the results obtained using Alternative 1 for the case in which the state \(x = -1.00\). Clearly, different measurement and input binary data sequences are used than in the previous two figures. These two sequences are not explicitly listed because they do not provide valuable information. After fifty iterations the state estimate \(\hat{x}(k/k)\) is -1.18 with an RMS error over fifty iterations of .18. However, the data bit sequence estimate \(\{\hat{u}(k)\}\) is unsatisfactory. The number of detection errors is 10, which is far too many. Earlier it was pointed out that a large number of detection errors should generally be expected for very low SNR cases. Clearly \(x = -1.00\) falls in the very low SNR class. We have encountered for the first time the situation which was considered in Chapter 3, in which the AJDEA provides a "good" state estimate but an unsatisfactory binary sequence estimate \(\{\hat{u}(k)\}\). For these situations it was suggested that an extra step be performed. Basically using
Fig. 4.4 \( x = -1.00 \)

Alternative 1
Fig. 4.5 \( x = -1.00 \)

Alternative 2
\( \hat{x}(50/50) \) as the state, the Viterbi algorithm can be used to generate the "optimal" binary sequence estimate. It is important to point out that this sequence estimate is "optimal" only in the sense that we used the best available estimate at the end of the transmission interval, i.e., \( \hat{x}(50/50) \).

On the other hand, Figure 4.5 is the representation of the results obtained using Alternative 2 for the same case as in Figure 4.4, i.e., the same input sequence \( \{u(k)\} \) and measurement sequence \( \{z(k)\} \) is used and the state \( x \) is -1.00. After fifty iterations the state estimate \( \hat{x}(50/50) \) is -1.45 with a RMS error over fifty iterations of .32. Once again, by comparing the RMS errors of the two Alternatives (Figures 4.4 and 4.5), we conclude that Alternative 1 (maximum likelihood detection) generates a better state estimate than Alternative 2 (weighted residual estimation).

4.5 Summary

In this chapter a number of key issues of the Adaptive Joint Detection Estimation Algorithm were studied by examining numerous computer simulation results of the Scalar Case.

It was emphasized earlier that the major objective in this thesis was to formulate an adaptive and high-throughput procedure for simultaneously generating the state estimate and detecting the transmitted data bit sequence. The numerical results presented in this chapter have demonstrated that this objective has successfully been attained with the
AJDEA for the simple Scalar Case. It was shown that one initialization bit was necessary to ensure proper operation of the algorithm. Moreover, it was pointed out that for both Alternatives the throughput of the algorithm is given by \( \frac{N}{N+1} \), where \( N \) is the length of the actual data transmission. For reasonable bursts of data transmission the algorithm provides high-throughput.

Finally, it was shown using simulation results that, in general, Alternative 1 had smaller estimation errors than Alternative 2. More will be said about this crucial issue in the ensuing chapters. In fact, it is shown in Chapter 6 that, for more complex communication systems, Alternative 2 can actually outperform Alternative 1.
CHAPTER 5

DISCUSSION OF THE TWO-DIMENSIONAL VECTOR CASE

5.1 Introduction

The Two-Dimensional Vector Case is the second of the two preliminary cases that are investigated in this thesis report. It is felt that by first studying the application of the Adaptive Joint Detection Estimation Algorithm to the Scalar Case and the Two-Dimensional Vector Case, insight will be gained that will prove to be invaluable in studying the Multipath Communication Case in Chapter 6.

In Section 5.2 the general vector equations of the AJDEA are adapted for the Two-Dimensional Vector Case. Section 5.3 presents the results obtained from the application of the AJDEA to various different examples. In Sections 5.4 and 5.5 unexpected step changes of the state vector of the system are introduced. Computer simulations demonstrate the robustness of the algorithm in successfully coping with these step jumps. Finally in Section 5.5 a brief summary of the major issues elucidated in this chapter is given.

The Two-Dimensional Vector Case should be regarded as a contrived intermediate example used to understand better the operation of the AJDEA before proceeding to
the Multipath Communication Case. Actually, by studying the problem definition of the system given below it is apparent that it can be thought of as a greatly simplified version of the Multipath Communication Case.

The Two-Dimensional Vector system is defined by the following pair of discrete state space equations:

**State Dynamics**

\[ x(k+1) = x(k) \]  \hspace{1cm} (5.1)

**Measurement Equation**

\[ z(k) = C(k)x(k) + v(k) \]  \hspace{1cm} (5.2)

where \( x(k) \) is a two-dimensional state vector, \( z(k) \) is the scalar measurement vector. The noise source \( v(k) \) is assumed to be a zero mean white Gaussian noise of covariance \( R \). Note that the system has no process noise, i.e., the state \( x(k) \) is constant over time. The initial state estimate \( \hat{x}(0/0) \) is assumed to be Gaussian with mean \( x_0 \) and covariance \( P(0/0) \).

The observation matrix \( C(k) \) is an unknown matrix given by

\[ C(k) = [u(k) \ u(k-1)] \]  \hspace{1cm} (5.3)

where \( u(k) \) is the transmitted data bit that is equal to \( \pm 1 \) with equal a priori probability. Thus, \( C(k) \) is a \((1x2)\) matrix.
whose first element is the most recent data bit transmitted over the channel, while the second element is the preceding data bit. Moreover, the input sequence \( \{u(k)\} \) is white, i.e., \( u(k) \) and \( u(j) \) are uncorrelated for \( k \neq j \).

Once again, as in the Scalar Case, the major objective is to obtain simultaneously adaptive estimates of the input binary sequence \( \{u(k)\} \) and the state vector \( \mathbf{x}(k) \) with the highest throughput possible. The procedure that is used in attaining this objective is the Adaptive Joint Detection Estimation Algorithm (AJDEA) of Chapter 3.

5.2 Adaptation of the AJDEA to the Two-Dimensional Vector Case

It has already been emphasized that its state dynamics are not driven by process noise. However, it shall be shown in the next section that for proper operation of the algorithm artificial Gaussian process noise with zero mean and \( (2 \times 2) \) covariance matrix \( \mathbf{Q} \) should be added.

The general vector equations of the AJDEA (3.11) through (3.15) reduce to the following set of equations:

**Predict Cycle**

\[
\hat{\mathbf{x}}(k/k-1) = \hat{\mathbf{x}}(k-1/k-1) \\
\mathbf{P}(k/k-1) = \mathbf{P}(k-1/k-1) + \mathbf{Q}
\]
Update Cycle

\[
\hat{x}(k/k) = \hat{x}(k/k-1) + G(k)[z(k) - \hat{C}(k)\hat{x}(k/k-1)] + G(k)[z(k) - C(k)x(k/k-1)] \\
\Sigma(k/k) = [I - G(k)\hat{C}(k)]\Sigma(k/k-1) \\
G(k) = \Sigma(k/k-1)\hat{C}^T(k)[\hat{C}(k)\Sigma(k/k-1)\hat{C}^T(k) + R]^{-1}
\]

where \(G(k)\) is a \((2 \times 1)\) gain matrix and \(\Sigma(k/k)\) is a \((2 \times 2)\) covariance matrix.

The next step is the adaptation of the parameter identification procedure of the AJDEA to the Two-Dimensional Vector Case. The same definition for the two hypotheses \(H_0\) and \(H_1\) is assumed as in the "limited" general case of Chapter 3, i.e., \(H_0\) is the hypothesis that \((u(k)=-1)\), while \(H_1\) is the hypothesis that \((u(k)=1)\).

In order to construct the observation matrix estimate \(\hat{C}(k)\), the data bit estimate \(\hat{u}(k)\) must be generated. Once again as in the previous chapter, two distinct procedures (Alternative 1 and Alternative 2) are used to furnish two binary data bit estimates respectively. From Equation (3.22) we know that

**Alternative 1: Maximum Likelihood Detection of \(u(k)\)**

\[
\text{choose } H_1: \hat{u}(k) = 1 \\
p(z(k)/H_1, z(k-1)) > p(z(k)/H_0, z(k-1)) \\
H_0: \hat{u}(k) = -1
\]
where \( p(z(k)/H_j,Z(k-1)), j \in \{0,1\}, \) is a Gaussian density given by

\[
\exp\left\{-(z(k)-\bar{z}(k|k-1))^2/2(\bar{z}(k|k-1)\bar{z}(k|k-1)^T+R)\right\}
\]

\[
(2\pi)^{\frac{1}{2}}[\bar{z}(k|k-1)\bar{z}(k|k-1)^T+R]^{-\frac{1}{2}}
\]

(5.10)

and

\[
Z(k) = \{z(1), z(2), z(3), \ldots, z(k)\}.
\]

(5.11)

In Chapter 3 (Property 4) \( C_1(k) \) and \( C_0(k) \) were defined to be matrices that correspond respectively to the two models (hypotheses) \( H_0 \) and \( H_1 \). Under hypothesis \( H_0 \) all the elements of \( C(k) \) that are equal to \( u(k) \) are set to -1, while under \( H_1 \) they are set to +1. Therefore

\[
C_1(k) = \begin{bmatrix} 1 & \hat{u}(k-1) \end{bmatrix}
\]

(5.12)

\[
C_0(k) = \begin{bmatrix} -1 & \hat{u}(k-1) \end{bmatrix}
\]

(5.13)

where \( \hat{u}(k-1) \) is the estimate of the bit \( u(k-1) \) generated by the AJDEA - Alternative 1 at time \( k-1 \). Once \( \hat{u}(k) \) has been evaluated, the observation matrix estimate \( \hat{C}(k) \) can be constructed, i.e.,

\[
\hat{C}(k) = \begin{bmatrix} \hat{u}(k) & \hat{u}(k-1) \end{bmatrix}
\]

(5.14)
Alternative 2: Weighted Residual Estimation of $u(k)$

\[
\hat{u}(k) = P_1(k) [1] + P_0(k) [-1] = P_1(k) - P_0(k) \tag{5.15}
\]

where $P_1(k)$, $(i \in \{0,1\})$, is the a posteriori probability that the $i$th model matches the actual system (Equation (3.7)). $P_1(k)$ is given by the following equation

\[
P_1(k) = \frac{1}{\sum_{j=0}^{1} p(z(k)/H_j, Z(k-1))} \tag{5.16}
\]

where $p(z(k)/H_1, Z(k-1))$ is the Gaussian density given by Equation (5.10).

Once the data bit estimate $\hat{u}(k)$ has been evaluated, the observation matrix estimate $\hat{c}(k)$ can be constructed, i.e.,

\[
\hat{c}(k) = [\hat{u}(k) \quad \hat{u}(k-1)] \tag{5.17}
\]

Earlier in this chapter it was indicated that the initial state estimate $\hat{x}(0/0)$ is assumed to be Gaussian with mean $x_0$ and covariance $\Sigma(0/0)$. However, to insure the proper operation of the Adaptive Joint Detection Estimation Algorithm, further restrictions on the mean and covariance of $\hat{x}(0/0)$ have to be imposed. In particular,

1) the mean $x_0$ must be set to zero and the first bit $u(1)$ must be an initialization bit.

2) the covariance $\Sigma(0/0)$ must be a diagonal matrix.
In order to avoid unnecessary repetition, the reader is referred to Equation (4.24) and the ensuing discussion for an elucidation of the first restriction. For example, a computer simulation was performed for the case \( x = [3, 5] \) and \( x_0 = [-4, 0] \). The algorithm never operated properly. After fifty increments \( \hat{x}(50/50) = [-5.3, .35] \) and 27 detection errors were made. Note that \( \hat{x}(50/50) \) is not approximately equal to the negative of \( x \) as would have been expected in the Scalar Case. This is due to the structure of \( C(k) \).

In the vector case the measurements \( z(k+1) \) and \( z(k) \) are correlated since \( C(k+1) \) and \( C(k) \) share a common element \( u(k) \).

However, the second restriction was not an issue in Chapter 4 and thus, now will be discussed in detail.

From Equations (5.2) and (5.3) it can be shown that

\[
z(1) = \begin{bmatrix} u(1) & 0 \end{bmatrix} \begin{bmatrix} x_1(1) \\ x_2(1) \end{bmatrix} + v(1) \tag{5.18}
\]

where \( x_1 \) and \( x_2 \) are the elements of the state vector \( x \) and all bits preceding \( u(1) \) are set to zero, i.e., \( u(i) = 0 \) for \( i < 1 \). Moreover, since it has been assumed that \( u(1) = 1 \), then

\[
z(1) = x_1(1) + v(1) \tag{5.19}
\]

Note in particular that no information concerning \( x_2(1) \) is contained in measurement \( z(1) \). Therefore, the estimator should only update the estimate of \( x_1 \), and \( \hat{x}_2(1/1) \) should
remain equal to \( \hat{x}_2(0/0) \), i.e., \( \hat{x}_2(0/0) = 0 \).

Now let the initial covariance \( \Sigma(0/0) \) be given by

\[
\Sigma(0/0) = \begin{bmatrix}
\Sigma_{11}(0) & \Sigma_{12}(0) \\
\Sigma_{12}(0) & \Sigma_{22}(0)
\end{bmatrix}
\]  

(5.20)

since a covariance matrix must be symmetric, i.e., \( \Sigma(0/0) = \Sigma^T(0/0) \).

By using Equations (5.6) and (5.8) it can easily be shown that

\[
\hat{x}(1/1) = G(1)z(1) = \begin{bmatrix} G_1(1) \\
G_2(1) \end{bmatrix} z(1)
\]  

(5.21)

and

\[
G(1) = \begin{bmatrix}
\Sigma_{11}(0)/(\Sigma_{11}(0) + R) \\
\Sigma_{12}(0)/(\Sigma_{11}(0) + R)
\end{bmatrix}
\]  

(5.22)

Clearly from Equation (5.21), \( \hat{x}_2(1/1) \) can only be zero if \( G_2(1) \) equals zero. Hence, from Equation (5.22) it is evident that the off-diagonal term should be zero, i.e.,

\[
\Sigma_{12}(0) = 0
\]  

(5.23)

and thus \( \Sigma(0/0) \) is a diagonal matrix.

Therefore, we have shown that if \( \Sigma(0/0) \) has nonzero off diagonal terms, then the estimator generates an estimate
\hat{x}_2(1/l) based on a measurement z(1) that contains no information about \(x_2(l)\). Computer simulation results have demonstrated that this can cause serious estimation errors. For example, consider the following case:

\[
\begin{pmatrix}
2.0 \\
-2.0
\end{pmatrix}; \quad \begin{pmatrix}
4 \\
2 \\
4
\end{pmatrix}.
\]

The AJDEA furnished the state estimate after fifty iterations:

\[
\hat{x}(50/50) = \begin{pmatrix}
2.97988 \\
.45865
\end{pmatrix}
\]

which is clearly an unsatisfactory result.

We have completed the discussion of the two necessary restrictions on the initial state estimate \(\hat{x}(0/0)\). The adaptation of the AJDEA to the Two-Dimensional Vector Case is also complete. In the next section computer simulation results for various state variable values are presented.

Therefore, for both familiarizing the reader with the sequence of steps used in implementing the algorithm and summarizing the results of this section, the following list of the sequence of steps is given:

\[
i=0 \text{ Initial Conditions} \\
1) \quad \hat{x}(0/0) = x_0 = 0
\]
2) $\Sigma(0/0) = \begin{bmatrix} \Sigma_{11}(0) & 0 \\ 0 & \Sigma_{22}(0) \end{bmatrix}$ (5.27)

$\Sigma_{11}(0) > 0, \Sigma_{22}(0) > 0$

$i = 1$

1) Set $u(1) = 1$
2) Calculate $G(1)$
3) Compute covariance matrix $\Sigma(1/1)$
4) Update state estimate

$$\hat{x}(1/1) = G(1)z(1)$$ (5.28)

$i = k$

1) Evaluate $\hat{u}(k)$
2) Calculate $G(k)$
3) Compute $\Sigma(k/k)$
4) Update state estimate

$$\hat{x}(k/k) = \hat{x}(k/k-1)$$

$$+ G(k)[z(k) - (\hat{u}(k) - \hat{u}(k-1))\hat{x}(k/k-1)]$$ (5.29)

5.3 Computer Simulation Results

In this section the computer simulation results of the Two-Dimensional Vector Case under both "high to moderate" and "low" signal to noise ratios (SNR's) using both Alternative 1 and Alternative 2 are presented.

The measurement sequence $\{z(k)\}$ is generated by first using a random number generator with different seeds to generate the transmitted bit sequence $\{u(k)\}$ and a Gaussian distributed
noise sequence \( \{v(k)\} \) with zero mean and unit variance. Then by selecting a value for the state \( x \) and implementing Equation (5.2), the measurement sequence \( \{z(k)\} \) is generated.

In Table 5.1 all the simulation results that are discussed in this section are listed for quick reference with their associated figure numbers. The casual reader can merely examine Table 5.1 and the figures and avoid the detailed discussion of the results presented in this section.

"High to Moderate" Signal to Noise Ratio (SNR)

Under these SNR conditions the input data bit estimate \( u(k) \) generated by the weighted residual approach (Alternative 2) is essentially the same as the one furnished by Alternative 1. This is due to the fact that under high SNR conditions the a posteriori probabilities \( P_1(k) \) and \( P_0(k) \) are very close to 0 and 1 or vice versa. Intuitively this implies that given a measurement that is barely corrupted with noise one can determine almost unequivocally which is the actual model (hypothesis). Therefore, since under Alternative 1 only the maximum likelihood hypothesis is selected, i.e., \( P_1(k) \) and \( P_0(k) \) are equal to 0 and 1 or vice versa, then \( \{\hat{u}(k)\} \) generated by both Alternatives are almost identical. Consequently, since the two bit sequence estimates \( \{\hat{u}(k)\} \) are almost the same, the state estimate \( \hat{x}(k/k) \) generated by both Alternatives are approximately equal. Extensive computer simulations were performed that attest this assertion.
<table>
<thead>
<tr>
<th>DESCRIPTION</th>
<th>$\hat{x}_1(30/50)$</th>
<th>$\hat{x}_2(50/50)$</th>
<th>RMS ERROR OF $\hat{x}_1(k/k)$</th>
<th>RMS ERROR OF $\hat{x}_2(k/k)$</th>
<th>NUMBER OF DETECTION ERRORS</th>
<th>FIGURES</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x = [4.0$ $-5.0]^T$</td>
<td>4.5</td>
<td>-4.3</td>
<td>1.1</td>
<td>2.2</td>
<td>2</td>
<td>5.1</td>
</tr>
<tr>
<td>no artificial noise; $\sigma = 0$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$x = [4.0$ $-5.0]^T$</td>
<td>4.5</td>
<td>-4.9</td>
<td>.96</td>
<td>2.1</td>
<td>2</td>
<td>5.2</td>
</tr>
<tr>
<td>artificial noise added; $\sigma = .010$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$x = [4.0$ $-5.0]^T$</td>
<td>3.3</td>
<td>-4.8</td>
<td>.61</td>
<td>1.1</td>
<td>--</td>
<td>5.3</td>
</tr>
<tr>
<td>artificial noise added; known learning sequence</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$x = [2.0$ $-1.5]^T$</td>
<td>2.1</td>
<td>-1.3</td>
<td>.34</td>
<td>.79</td>
<td>1</td>
<td>5.4</td>
</tr>
<tr>
<td>Alternative 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$x = [2.0$ $-1.5]^T$</td>
<td>2.3</td>
<td>-1.5</td>
<td>.57</td>
<td>.47</td>
<td>1</td>
<td>5.5</td>
</tr>
<tr>
<td>Alternative 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$x = [-2.25$ $2.25]^T$</td>
<td>2.2</td>
<td>2.1</td>
<td>.35</td>
<td>.61</td>
<td>0</td>
<td>5.6</td>
</tr>
<tr>
<td>Alternative 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$x = [-2.25$ $2.25]^T$</td>
<td>2.3</td>
<td>2.2</td>
<td>.43</td>
<td>.57</td>
<td>0</td>
<td>5.7</td>
</tr>
<tr>
<td>Alternative 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$x = [-1.75$ $1.75]^T$</td>
<td>-1.98</td>
<td>1.74</td>
<td>.19</td>
<td>.32</td>
<td>5</td>
<td>5.8</td>
</tr>
<tr>
<td>Alternative 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>seed = 78654</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$x = [-1.75$ $1.75]^T$</td>
<td>-2.13</td>
<td>1.89</td>
<td>.38</td>
<td>.33</td>
<td>5</td>
<td>5.9</td>
</tr>
<tr>
<td>Alternative 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>artificial noise added</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$x = [-1.75$ $1.75]^T$</td>
<td>-2.15</td>
<td>.94</td>
<td>.72</td>
<td>1.4</td>
<td>10</td>
<td>--</td>
</tr>
<tr>
<td>Alternative 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>seed = 7865</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$x = [-1.75$ $1.75]^T$</td>
<td>-2.32</td>
<td>.89</td>
<td>.91</td>
<td>1.2</td>
<td>11</td>
<td>--</td>
</tr>
<tr>
<td>Alternative 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>seed = 7865</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>artificial noise added</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**TABLE 5.1**

COMPUTER SIMULATION RESULTS OF TWO-DIMENSIONAL CASE
Earlier in this chapter it was pointed out that since the system is not driven by a plant noise sequence, an artificial Gaussian noise sequence is added in order to enhance the performance of the algorithm. Figure 5.1 and Figure 5.2 taken together confirm this fact. Figure 5.1 is an example of the state estimate generated by the AJDEA - Alternative 2 with no artificial noise, i.e., $Q = \mathbf{0}$, for the case in which $\mathbf{x} = [4.0 \quad -5.0]^T$. Note that the estimate $\hat{x}(k/k)$ never actually reaches the actual state value. Essentially the algorithm becomes very confident with its estimates after approximately twenty iterations and thereafter runs open loop and ignores the remaining measurements. Clearly, even though the measurements are corrupted with noise they still contain "useful" information and should be utilized by the filter.

The above intuitive argument can be strengthened by examining the state estimation equation (5.6) which is written below

$$\hat{x}(k/k) = \hat{x}(k/k-1) + Q(k)[z(k)-\hat{z}(k)\hat{x}(k/k-1)]$$

(5.30)

The above behavior, whereby the filter runs open loop, is attributed to the general decrease of the magnitude of the elements of $Q(k)$. This decrease can be seen by studying Equations (5.5), (5.7), and (5.8).
Fig. 5.1 \( \mathbf{x} = [4.0 \ -5.0]^T \)
No Process Noise
Fig. 5.2 $x = [4.0 \ -5.0]^T$

Process Noise Added
Hence, the gain matrix must be maintained. This can be done by adding artificial white Gaussian noise to the covariance matrix in the update cycle, i.e.,

\[ \Sigma(k/k-1) = \Sigma(k/k) + Q \]  

(5.31)

Figure 5.2 is the same case as Figure 5.1, only the system is driven by an artificial Gaussian noise with covariance \( Q \) given by

\[ Q = \begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix} \]  

(5.32)

The final state estimate \( \hat{x}(50/50) \) obtained is \([4.45, -4.87]^T\). In particular, note the substantial decrease in the average RMS error for both elements of the vector in Figure 5.2 as compared to Figure 5.1. However, it is important to emphasize that there can often be undesirable side effects in adding artificial noise. The addition of the process noise increases the bandwidth of the filter, thus permitting a greater amount of measurement noise to filter through, which can ultimately increase the RMS errors of the estimates. Therefore, caution
and good engineering judgement should be employed in determining which value to use for the covariance matrix $Q$. More will be said about this topic in Section 5.4 in which step changes in the state vector are introduced. Furthermore, hereafter all simulations in this chapter are made with $Q = \begin{pmatrix} 0.01 & 0 \\ 0 & 0.01 \end{pmatrix}$.

The data sequence estimate $\{\hat{u}(k)\}$ is not given in Figure 5.2 because it is immaterial. The only important result is the number of data bit detection errors which is given in the upper right corner of Figures 5.1 and 5.2. Note that for both figures the number of errors in detection is two.

Figure 5.3 is once again concerned with the same case as in the previous cases, namely $x = \begin{pmatrix} 4.0 \\ -5.0 \end{pmatrix}^T$. However, instead of a random data sequence a known learning sequence is used, i.e., there is no longer an unknown parameter sequence to detect. Thus, the Adaptive Joint Detection Estimation Algorithm reduces to a standard Kalman filter and $\hat{x}(k/k)$ plotted in Figure 5.3 is the optimal (minimum mean square error) estimate of the state vector. Hence, Figure 5.3 provides a lower bound on the root mean square (RMS) error of $\hat{x}(k/k)$ over fifty increments for this particular case.

The two detection errors in both Figure 5.1 and Figure 5.2 occur at $k = 2$ and $k = 8$, which explains the large estimation errors of both state vector elements for the first twenty time increments. However, note that for the remaining
Fig. 5.3 $\mathbf{x} = [4.0 \ -5.0]^T$

Learning Sequence
time increments the state estimates in Figure 5.2 are very close to their corresponding optimal estimates in Figure 5.3.

Moreover, since the state estimate is "good" the probability of having a detection error is small. What we have just encountered is a cyclic property of the AJDEA that is due to the direct coupling that exists between the detection and estimation procedures. (See Figure 1.1). When the RMS error of the state estimate is small, the probability of making a detection error is small.

Based on the preceding discussion and other extensive computer simulations, we can claim that the AJDEA estimates of both the state vector and binary input sequence are extremely satisfactory. The next step is to investigate the performance of the AJDEA under lower signal to noise ratio conditions.

"Low" Signal to Noise Ratio

As expected, the Adaptive Joint Detection Estimation Algorithm does not generally perform as well under "low" SNR as it did under "high to moderate" SNR conditions. The number of detection errors and the estimation errors can generally be expected to increase. In this section computer simulation results implementing both Alternatives 1 and 2 are presented and compared.

Figure 5.4 is the presentation of the results obtained using Alternative 1 for the case in which the state vector is
[2.0 -1.5]^T. After fifty iterations the state vector estimate is [2.1 -1.3]^T with an RMS error over fifty iterations of .341 for the first element and .790 for the second element. The unknown data bit sequence estimate \{\hat{u}(k)\} is excellent with only one detection error made out of forty-nine transmitted bits.

On the other hand, Figure 5.5 is the presentation of the results obtained using Alternative 2 for the same case as Figure 5.4, i.e., the same input sequence \{u(k)\} and measurement sequence \{z(k)\} is used and the state vector is [2.0 -1.5]^T. After fifty iterations the state vector estimate is [2.27 -1.51]^T with a root mean square error over fifty iterations of .574 for the first element \(x_1(k)\) and .473 for the second element \(x_2(k)\). By comparing the root mean square errors of the two Alternatives, one concludes that Alternative 2 generates a better estimate of \(x_2(k)\), i.e., the second element of the state vector, while Alternative 1 generates a better estimate of \(x_1(k)\). The unknown data bit sequence estimate \{\hat{u}(k)\} is excellent with only one error made in identifying forty-nine transmitted bits. Actually, for both Figures 5.4 and 5.5, the detection error occurred at \(k = 2\), i.e., \(\hat{u}(2) \neq u(2)\).

The next case that is considered is depicted in Figures 5.6 and 5.7. Figure 5.6 is the representation of the results obtained using Alternative 1 (i.e., maximum likelihood estimation
\[ x = [2.0, -1.5]^T \]
Alternative 1
Fig. 5.5 $x = [2.0 \ -1.5]^T$
Alternative 2
of \( u(k) \) for the case in which the state vector is \([-2.25, 2.25]^T\). Clearly, different measurement and input binary sequences are used than in the previous two figures. These two sequences are not explicitly listed because they do not provide valuable information. After fifty iterations the state vector estimate is \([-2.17, 2.08]^T\) with an RMS error over fifty iterations of \(0.350\) for the first element \(x_1(k)\) and \(0.611\) for the second element \(x_2(k)\). Moreover, the unknown data bit sequence estimate \(\{\hat{u}(k)\} \) is excellent with no detection errors made in identifying forty-nine unknown transmitted bits.

On the other hand, Figure 5.7 is the representation of the results obtained using Alternative 2 for the same case as in Figure 5.6, i.e., the same input sequence \(\{u(k)\}\) and measurement sequence \(\{z(k)\}\) is used and the state vector is \([-2.25, 2.25]^T\). Once again it is important to emphasize that these two sequences for Figures 5.6 and 5.7 are different and independent from those for Figures 5.4 and 5.5. After fifty iterations the state vector estimate is \([-2.34, 2.17]^T\) with an RMS error over fifty iterations of \(0.429\) for the first element and \(0.568\) for the second element. Once again, by comparing the RMS errors of the two Alternatives, one concludes that Alternative 2 generates a better estimate of \(x_2(k)\), while Alternative 1 generates a better estimate of \(x_1(k)\). The unknown data bit sequence estimate \(\{\hat{u}(k)\} \) is excellent with
Fig. 5.6 $x = [-2.25, 2.25]^T$

Alternative 1
Fig. 5.7 $\hat{x} = [-2.25, 2.25]^T$

Alternative 2
once again no detection errors made in identifying forty-nine unknown transmitted bits.

Actually, thus far the two cases considered in Figures 5.4, 5.5 and Figures 5.6, 5.7, respectively, have a better overall performance than the higher SNR case in Figure 5.2. However, these two cases gave unusually good results for "low" SNR conditions. In general, one will find the performance of the AJDEA under such low SNR conditions unpredictable and sensitive to various parameters. For example, the particular data sequence transmitted does not affect the RMS error performance of the state estimator under high SNR conditions. However, it was demonstrated that by using different input sequences for the case \( x = [-1.75 \ 1.75] \), markedly significant differences in performance were obtained.

First we consider the case in which the AJDEA performed better. Figure 5.8 is the representation of the results obtained using Alternative 1 for the case in which the state vector is \([-1.75 \ 1.75]^T\) and (78654) was the seed that was used to initiate the random number generator that generated the fifty bit input sequence. After fifty iterations the state vector estimate is \([-1.98 \ 1.74]^T\) with an RMS error over fifty iterations of .189 for the first element \( x_1(k) \) and .318 for the second element \( x_2(k) \). Finally, the unknown data bit sequence estimate \( \hat{u}(k) \) is generated with five detection errors made in identifying forty-nine unknown transmitted
Fig. 5.8 $\mathbf{x} = [-1.75, 1.75]^{T}$

Alternative 1
bits.

On the other hand, Figure 5.9 is the representation of the results obtained using Alternative 2 for the same case as in Figure 5.8, i.e., the same input and measurement sequences are used and the state vector is \([-1.75 \ 1.75]^T\). After fifty iterations the state vector estimate is \([-2.13 \ 1.89]\) with an RMS error over fifty iterations of \(.380\) for the first element \(x_1(k)\) and \(.33\) for the second element \(x_2(k)\). The unknown data bit estimate \(\{\hat{u}(k)\}\) is generated with five detection errors made in identifying forty-nine unknown transmitted bits.

Earlier it was claimed that the performance of the AJDEA under low SNR can actually depend on the particular input data bit sequence used. This claim is verified by the following simulation results. The same case as in Figures 5.8 and 5.9 was reconsidered with a different input binary sequence (seed = 7865), \(x = [-1.75 \ 1.75]^T\). Using Alternative 1 the state vector estimate after fifty iterations was \([-2.15 \ .84]^T\) with a RMS error over fifty iterations of \(.717\) for the first element \(x_1(k)\) and \(1.39\) for the second element \(x_2(k)\). Moreover, the unknown data bit sequence estimate \(\{\hat{u}(k)\}\) is unsatisfactory with ten detection errors made in identifying forty-nine unknown transmitted bits.

Similarly, using Alternative 2 with the second input data bit sequence (i.e., seed = 7865) the AJDEA performance
Fig. 5.9 \( \mathbf{x} = [\begin{array}{c} -1.75 \\ 1.75 \end{array}]^T \)

Alternative 2
results were unsatisfactory. After fifty iterations the state vector \( x = [-1.75 \ 1.75]^\top \) estimate was \([-2.32 \ 0.89]^\top \) with an RMS error over fifty iterations of 0.914 for the first element \( x_1(k) \) and 1.15 for the second element. The unknown data bit estimate is unsatisfactory with eleven detection errors made in identifying forty-nine unknown transmitted bits.

Thus far, three illustrative cases of low SNR conditions have been discussed. Simulations were performed for a large number of other cases with very similar results. To avoid redundancy these results are not presented in this report.

However, we conclude that based upon all the results for "low" SNR conditions, the performance of the AJDEA is still satisfactory. In general, Alternative 1 provided a slightly better overall performance than Alternative 2. However, this difference was not as pronounced as for the Scalar Case in which Alternative 1 performed much better than Alternative 2. At this stage no clear-cut, indisputable explanation for this phenomenon can be given. One possible argument is that for the vector case the measurement sequence \( z(k) \) is no longer uncorrelated. (The observation matrix \( C(i) \) and \( C(i+1) \) have in common the element \( u(i) \)). Further discussion of the topic is presented in the next chapter in which the more complex Multipath Communication Case is considered. An interesting question to raise at this point is
whether or not the trend will continue and perhaps Alternative 2 (weighted residual estimation) may even perform better than Alternative 1 (maximum likelihood detection).

5.4 Special Cases with Step Changes of the State Vector

This section of the chapter first provides motivation for studying Two-Dimensional Vector Cases that experience unexpected step changes in the state vector. The next step is to adjust the dynamic system Equations (5.1) and (5.2) to encompass a larger selection of "Two-Dimensional" systems; in particular to include systems that experience step changes of the state vector. Illustrative computer simulation results are presented and discussed in the next section.

Earlier in this chapter it was emphasized that the Two-Dimensional Vector Case was devised to serve as an intermediate step with the Multipath Communication Case as an ulterior motive. It should be viewed as a mathematical model that is used to gain insight into the operation of the Adaptive Joint Detection Estimation Algorithm, which will prove to be very valuable in the analysis of the Multipath Communication Case. Therefore, we refrain from associating particular physical characteristics to the Two-Dimensional Vector Case and focus our attention on its mathematical properties.

The motivation for introducing step changes of the
state vector is to further test the robustness of the Adaptive Joint Detection Estimation Algorithm. The algorithm should be able to recover from sudden step changes of any one of the elements of the state vector and quickly track the new values. It was primarily because of this reason that we insisted earlier on the addition of artificial white noise for maintaining the magnitude of the gains of the estimator. If the gain magnitude is allowed to decrease rapidly, the estimator eventually runs open loop and ignores future measurements. Clearly, under such conditions the estimator cannot rapidly react to the sudden step changes of the state.

An alternative procedure that we could have used instead of artificial noise is an on-line adaptive scheme which constantly monitors the behavior of the residuals of the estimator. In Appendix B (Equation (5.11)) the residual \( r(k) \) was defined as the difference between the actual measurement at time \( k \) and the predicted measurement, i.e.,

\[
r(k) = z(k) - \hat{x}(k|k-1) \tag{5.33}
\]

If the magnitude of the residuals suddenly increases for a number of consecutive increments, then the error covariance matrix \( \Sigma(k/k) \) is temporarily "boosted up," which in turn increases the magnitude of the gain matrix \( G(k) \).

Essentially the same results can be obtained by using either procedure. However, we feel that for the cases of interest
in this report the artificial noise procedure is more appropriate.

The next step is the modification of the dynamic state Equations (5.1) and (5.2) to account for the sudden step changes that may occur. This situation can easily be modeled by adding an extra term to the state dynamics equation.

**State Dynamics**

\[ x(k+1) = x(k) + \alpha \delta(k,j) \quad (5.34) \]

where \( \alpha \) is an **unknown** (2x1) vector which represents the magnitude of the change in the two state vector elements. \( \delta(k,j) \) is the Kronecker delta function given by

\[ \delta(k,j) = \begin{cases} 1 & \text{if } k=j \\ 0 & \text{if } k\neq j \end{cases} \quad (5.35) \]

where \( j \in \{1, 2, 3, \ldots\} \) is the unknown time increment which denotes when the step change occurs. The measurement Equation (5.2) remains unaffected, i.e.,

**Measurement Equation**

\[ z(k) = C(k)x(k) + v(k) \quad (5.36) \]

By allowing unexpected step changes of the state vector the total uncertainty of the system has increased. It is not known when during the duration of the data transmission the
step change will occur or if it will occur at all. If a step change does occur, the magnitude is unknown.

Computer simulation results are presented in the next section to demonstrate that even with this added uncertainty the AJDEA still performs well. The detector still generates a data bit sequence estimate \( \hat{u}(k) \) with low detection errors and the estimator a state vector estimate \( \hat{x}(k/k) \) that tracks step jumps.

5.5 Further Computer Simulation Results

All the cases considered in this section fall in the "high to moderate" SNR conditions. In Section 5.2 it was established that under such conditions the results obtained using either Alternative 1 or Alternative 2 are essentially identical. Alternative 2 was used to generate all the simulation results discussed in this section.

Figure 5.10 is the representation of the results obtained using Alternative 2 for the case in which the state originally is \([3.0 \ 8.0]^T\), but at \(k = 25\) jumps to and remains thereafter at \([7.0 \ 5.0]^T\). Clearly, it is important that the estimator reacts as quickly as possible to the sudden change in the behavior of the residuals. In other words, it is important that the gain \( g(k) \) be large when the jump occurs. This can be assured by carefully choosing a large enough artificial process noise covariance \( Q \). It has been pointed out that engineering judgement must be used in selecting the value
Fig. 5.10
Step Change from $\tilde{x} = [3.0 \ 8.0]^T$ to $[7.0 \ 5.0]^T$
of $Q$. In general the larger the noise covariance, the larger the average RMS errors of the state estimate. Thus, there is a tradeoff between responsiveness and average RMS error.

After extensive tests using several different values for $Q$ the following value was selected and used in generating all the results in this section

$$Q = \begin{bmatrix} .04 & 0 \\ 0 & .04 \end{bmatrix}$$

(5.37)

Note that as expected, the above value of $Q$ is larger than the one in Equation (5.33), since unexpected step changes of the state vector may now occur.

Now returning back to Figure 5.10 we see that after twenty-four iterations the state vector estimate is $[3.35 \ 7.35]^T$. Immediately at $k = 25$ the estimator begins tracking the step changes and by $k = 36$ has locked onto the new values of the state vector. After fifty iterations the state vector estimate is $[6.86 \ 4.58]^T$. The estimator has successfully tracked the step change of the state vector from $[3.0 \ 8.0]^T$ to $[7.0 \ 5.0]^T$. Moreover, even with the step change, the unknown data bit sequence estimate $\hat{\{u(k)\}}$ is still excellent with only one detection error made in identifying forty-nine unknown transmitted bits. Actually, the one error occurred at $k = 3$ before the step change.

Thus far, no restrictions have been made on the charac-
teristics of the step change. From the discussion presented thus far it is clear that the step change should hopefully not change the sign of the elements of a state vector since the estimator will not be able to track the new values. The same reasoning that was used in setting \( \hat{x}(0/0) \) to zero can be applied in asserting the above statement.

In Figure 5.11 the same system is considered as the one in Figure 5.10 but with a different input binary sequence \( \{u(k)\} \), different noise sequence \( \{v(k)\} \), and therefore a different measurement sequence \( \{z(k)\} \). Hence, once again the state vector originally is \( [3.0 \ 8.0]^T \) but at \( k = 25 \) it jumps to and remains thereafter at \( [7.0 \ 5.0]^T \). After twenty-four iterations the state vector estimate is \( [2.80 \ 7.94]^T \). Immediately at \( k = 25 \) the estimator begins tracking the step changes and by \( k = 36 \) has locked onto the new values of the state vector. After fifty iterations the state vector estimate is \( [6.86 \ 4.51]^T \). Moreover, even with the step change at \( k = 25 \), the unknown data bit sequence estimate \( \{\hat{u}(k)\} \) is still excellent with no detection errors made in identifying forty-nine unknown transmitted bits.

Figure 5.12 considers a different system than the one in the previous two figures. The state vector originally is \( [9.0 \ 4.0]^T \) but at \( k = 25 \) it jumps to and remains thereafter at \( [7.0 \ 5.0]^T \). Once again, a different bit sequence \( \{u(k)\} \) has been transmitted than in any of the previous cases. After
Fig. 5.11
Step Change from $x = [3.0 \ 8.0]^T$ to $[7.0 \ 5.0]^T$
Fig. 5.12

Step Change from $x = [9.0 \ 4.0]^T$ to $[7.0 \ 5.0]^T$
twenty-four iterations the state vector estimate is \([8.91 \ 4.09]^T\)
which is an excellent estimate of the state vector \([9.0 \ 4.0]^T\).
Immediately at \(k = 25\) the estimator begins tracking the step
changes and by \(k = 30\) has locked onto the new values of the
state vector. After fifty iterations the state vector estimate
is \([6.43 \ 4.93]^T\). Thus, the estimator has successfully
tracked the step change of the state vector from \([9.0 \ 4.0]^T\)
to \([7.0 \ 5.0]^T\). Moreover, even with the step change, the
unknown data bit sequence estimate \(\{\hat{u}(k)\}\) is still excellent
with no detection errors made in the identification of forty-nine unknown transmitted bits.

Figure 5.13 is the representation of the results of
another example in which the state vector originally is
\([5.0 \ 3.0]^T\), but at \(k = 25\) it jumps to and remains there-
after at \([7.0 \ 5.0]^T\). After twenty-four iterations the
state vector estimate is \([5.53 \ 2.95]^T\). By \(k = 27\) the
estimator has begun to track the sudden step changes and
by \(k = 40\) has locked onto the new values of the state vector
estimate is \([6.54 \ 5.17]^T\). Once again, even with the step
change in the state vector at \(k = 25\) the unknown data bit
sequence estimate \(\{\hat{u}(k)\}\) is still excellent with no detection
errors made in the identification of the forty-nine unknown transmitted bits.

Figure 5.14 is the last example that is considered in
this chapter. It is the representation of the results of
Fig. 5.13

Step Change from \( \mathbf{x} = [5.0 \ 3.0]^T \) to \( [7.0 \ 5.0]^T \)
Fig. 5.14

Step Change from $x = [4.0 \ 4.0]^T$ to $[7.0 \ 5.0]^T$
the case in which the state vector originally is \([4.0 \ 4.0]^T\), but at \(k = 25\) it jumps to and remains thereafter at \([7.0 \ 5.0]^T\). After twenty-four iterations the state vector estimate \(\hat{x}(k/k)\) is \([4.41 \ 3.99]^T\). By \(k = 27\) the estimator has begun to track the sudden step changes and by \(k = 40\) has locked onto the new values of the state vector. After fifty iterations the state vector estimate is \([7.35 \ 5.35]^T\). Once again, even with the step changes at \(k = 25\) the unknown data bit sequence estimate \({\hat{u}(k)}\) is still excellent with no detection errors made in the identification of the forty-nine unknown transmitted bits.

There was a special reason for having chosen the vector \([4.0 \ 4.0]^T\) as the original state value. Note that the value of the two elements are equal. Therefore, whenever the observation matrix \(\mathbf{z}(k)\) is either \([1 \ -1]\) or \([-1 \ 1]\) the measurement will merely consist of measurement noise, i.e., \(z(k) = v(k)\). To illustrate this point the complete simulation results from which Figure 5.14 was formulated is given in Table 5.2. For example, note that for \(k = 6\) the measurement \(z(6)\) is equal to the measurement noise \(v(6)\). However, note that \(\hat{u}(6) = u(6)\), i.e., the detector did not make an error. As another example note that \(z(15)\), which is equal to \(0.14882\), consists only of noise. Once again, the detector operates properly and \(\hat{u}(15) = u(15)\). Moreover, note from these two examples that the sign of \(u(k)\) is not necessarily equal to
TABLE 5.2

STEP JUMP FROM $x = [4.0 \ 4.0]^T$ to $[7.0 \ 5.0]^T$
the sign of $z(k)$. For an elucidation of this issue the reader is referred to Chapter 3.

Aside from the examples discussed thus far in this chapter, a large number of other examples with both higher and lower signal to noise ratios were simulated with comparable results. Therefore, based on all these results we can conclude that the Adaptive Joint Detection Estimation Algorithm can endure a sudden step change in the state vector of the system and continue to generate an excellent bit sequence estimate $\{\hat{u}(k)\}$ and state vector estimate $\hat{x}(k/k)$. However, it should be pointed out that based on the results presented in Figure 5.10 through Figure 5.14 the interstice between consecutive step changes should be at least fifteen time increments in order to permit sufficient time for the estimator to lock onto the new values of the state vector.

5.6 Summary

In this chapter a number of key issues of the Adaptive Joint Detection Estimation Algorithm were studied by examining numerous computer simulation results of the Two-Dimensional Vector Case.

It was emphasized earlier that the major objective in this thesis was to formulate an adaptive and high-throughput procedure for simultaneously generating the state estimate and detecting the transmitted data bit sequence. The numerical results in this chapter have demonstrated that this objective
has successfully been attained with the AJDEA for the Two-
Dimensional Vector Case. In Section 5.2 it was concluded
that one initialization bit was necessary to ensure proper
operation of the algorithm. Therefore, the throughput of
the algorithm (defined in Equation (1.19)) is given by

\[
\text{Throughput} = \frac{A}{1 + N}
\]  

(5.37)

where \( N \) is the length of the actual data transmission.
Clearly, for reasonable values of \( N \) the algorithm provides
high-throughput. Moreover, in Section 5.4 the scope of the
Two-Dimensional Vector Case is extended to include systems
that can experience sudden step changes of the state vector.
Computer simulation results show that the AJDEA continues to
operate satisfactorily, thus manifesting the robustness of
the algorithm.
CHAPTER 6

6.1 Introduction

The Multipath Communication Case is the last, but most significant, of the three cases that are investigated in this thesis report. The case is based on the communication problem studied by Schneider and McGarty in "Reliable Satellite Communications for a Specular Multipath Channel" [29]. Specular multipath is prevalent in many mobile satellite communication systems and gives rise to distortions to the transmitted message sequence.

The specular multipath return can be characterized by a delay and a complex reflection coefficient. Moreover, it can be considered as intersymbol interference on the channel, which in turn can be modeled by a finite state machine.

In [29] Schneider and McGarty propose a suboptimal scheme for situations when complete knowledge of the multipath parameters is not readily available. Initially, a known data bit stream (learning sequence) is transmitted in order to obtain the state estimate of the channel. The technique employed is the standard Kalman filter. Then, by regarding the channel as a finite state machine, the Viterbi algorithm is implemented to extract the unknown data bits from
their respective received measurements.

In order to avoid unnecessary repetition a complete description of the physical model of the Multipath Communication Case is not provided. Instead at this point the reader is urged to review Section 1.1 of Chapter 1, where a detailed discussion of the physical model of the multipath system is given.

In Section 6.2 the general vector equations of the Adaptive Joint Detection Estimation Algorithm (Alternative 1 and Alternative 2) are adapted for the Multipath Communication Case. Section 6.3 discusses the results obtained from the application of the AJDEA to various different examples. In Sections 6.4 and 6.5 unexpected step changes of the state vector of the system are introduced. These step changes can be due to sudden step changes of the reflection coefficient, an unknown multipath parameter (random variable). In Chapter 1 it was characterized by a uniform density distributed between zero and one. Furthermore, computer simulation results demonstrate the robustness of the algorithm by successfully coping with these disturbances. Finally, in Section 6.6 a brief summary of the major issues elucidated in this chapter is provided.

The Multipath Communication Case is defined by the following pair of discrete state equations:
State Dynamics

\[ x(k+1) = x(k) \]  \hspace{1cm} (6.1)

Measurement Equation

\[ z(k) = C(k)x(k) + y(k) \]  \hspace{1cm} (6.2)

where \( x(k) \) is a 6-dimensional state vector, \( z(k) \) is the 2-dimensional measurement vector. The noise source \( y(k) \) is assumed to be zero mean white Gaussian noise of covariance \( R(2 \times 2) \). Note that once again the state dynamics are modeled as constant, i.e., no process noise. However, the definite need for additive artificial Gaussian process noise for such systems (i.e., not driven by process noise) was established in Chapter 5. Therefore, artificial white Gaussian noise is added to the system dynamics for all the computer simulations in this chapter so as to ensure the proper operation of the AJDEA. The covariance \( Q \) of the artificial noise essentially keeps the error covariances of the state estimate and the filter gains from "falling off" very fast. Otherwise, after a few iterations the filter would begin essentially running open loop and ignoring all future measurements.

The initial state estimate \( \hat{x}(0/0) \) is assumed to be Gaussian with mean \( \mu_0 \) and covariance \( P(0/0) \).

The observation matrix \( C(k) \) is an unknown matrix.
given by Equation (2.23), i.e.:

\[ z(k) = \sqrt{E} \left[ \begin{array}{cccc} u(k) & u(k-3) & u(k-4) & 0 & 0 & 0 \\ 0 & 0 & 0 & u(k) & u(k-1) & u(k-4) \end{array} \right] \]  

(6.3)

where \( E \) is the signal energy, \( u(k) \) is the data bit transmitted at time \( k \) and is equal to \( \pm 1 \) with equal a priori probability. Moreover, the input data sequence \( \{u(k)\} \) is white, i.e., \( u(k) \) and \( u(j) \) are uncorrelated for \( k \neq j \).

Once again, as in the previous two cases (the Scalar Case and the Two-Dimensional Vector Case), the major objective is to obtain simultaneously adaptive estimates of the input data sequence \( \{u(k)\} \) and the state vector \( x(k) \), with the highest throughput possible. The procedure that is used in attaining this objective is the Adaptive Joint Detection Estimation Algorithm (AJDEA) of Chapter 3.

6.2 Adaptation of the AJDEA to the Multipath Communication Case

It was indicated in Chapter 1 and Equation 6.1 that the multipath state dynamics are not driven by process noise. However, it shall be demonstrated in the next section that for proper operation of the algorithm artificial Gaussian noise should be added.

The general vector equations of the Adaptive Joint Detection Estimation Algorithm (3.11) through (3.15) reduce
to the following set of equations:

**Predict Cycle**

\[
\hat{X}(k/k-1) = \hat{X}(k-1/k-1) \quad (6.4)
\]

\[
\Sigma(k/k-1) = \Sigma(k-1/k-1) + Q \quad (6.5)
\]

**Update Cycle**

\[
\hat{X}(k/k) = \hat{X}(k/k-1) + G(k)[z(k)-\hat{C}(k)\hat{X}(k/k-1)] \quad (6.6)
\]

\[
\Sigma(k/k) = [I-G(k)\hat{C}(k)]\Sigma(k/k-1) \quad (6.7)
\]

\[
G(k) = \Sigma(k/k-1)\hat{C}^T(k)[\hat{C}(k)\Sigma(k/k-1)\hat{C}^T(k) + R]^{-1} \quad (6.8)
\]

where \(G(k)\) is a \((6x2)\) matrix and \(\Sigma(k/k)\) is a \((6x6)\) symmetric covariance matrix.

The adaptation of the estimator equations of the AJDEA for the Multipath Communication Case is complete. The next step is the adaptation of the parameter identification portion of the AJDEA. The same definition for the two hypotheses \(H_0\) and \(H_1\) is assumed as in the "limited" general case of Chapter 3, i.e., \(H_0\) is the hypothesis that \((u(k) = -1)\) while \(H_1\) is the hypothesis that \((u(k) = 1)\).

In order to construct the observation matrix estimate \(\hat{C}(k)\), the data bit estimate \(\hat{G}(k)\) must be generated. Once again as in the previous two chapters two distinct procedures

(Alternative 1 and Alternative 2) are used to furnish two binary data bit estimates respectively. From Equation (3.22) we know that

Alternative 1: Maximum Likelihood Detection of $u(k)$

choose $H_1: \hat{u}(k) = 1$

\[
p(z(k)/H_1, Z(k-1)) \frac{p(z(k)/H_0, Z(k-1))}{H_0: \hat{u}(k) = -1}
\]

where $p(z(k)/H_1, Z(k-1)), i \in \{0, 1\}$ is a Gaussian density equal to

\[
\frac{1}{(2\pi)^{m/2}[\det(C_1(k)\hat{\mathbf{x}}(k/k-1)C_1^T(k) + R)]^{1/2}}
\]

\[
C_1(k)\hat{\mathbf{x}}(k/k-1))^T(C_1(k)\hat{\mathbf{x}}(k/k-1)C_1^T(k) + R)^{-1}
\]

\[
\cdot (z(k) - C_1(k)\hat{\mathbf{x}}(k/k-1))
\]

where $m = \dim[z(k)]$ and

\[
Z(k) = \{z(1), z(2), \ldots, z(k)\}
\]

In Chapter 3 (Property 4) $C_1(k)$ and $C_0(k)$ were defined to be matrices that correspond respectively to the two models (hypotheses) $H_0$ and $H_1$. Under hypothesis $H_0$ all the elements of $C(k)$ that are equal to $u(k)$ are set to $-1$, while under $H_1$
they are set to +1. Therefore, the two matrices are given by

\[
C_1(k) = \begin{bmatrix}
1 & \hat{u}(k-3) & \hat{u}(k-4) & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & \hat{u}(k-1) & \hat{u}(k-4)
\end{bmatrix}
\]  \hspace{1cm} (6.13)

and

\[
C_0(k) = \begin{bmatrix}
-1 & \hat{u}(k-3) & \hat{u}(k-4) & 0 & 0 & 0 \\
0 & 0 & 0 & -1 & \hat{u}(k-1) & \hat{u}(k-4)
\end{bmatrix}
\]  \hspace{1cm} (6.14)

where \( \hat{u}(k-1), \hat{u}(k-3) \) and \( \hat{u}(k-4) \) denote the estimates generated by the AJDEA of the binary data bits \( u(k-1), u(k-3) \) and \( u(k-4) \) respectively.

Consequently, once \( \hat{u}(k) \) has been evaluated, the observation matrix estimate \( \hat{C}(k) \) is straightforwardly constructed by suitably altering either \( C_1(k) \) or \( C_0(k) \) as follows

\[
\hat{C}(k) = \begin{bmatrix}
\hat{u}(k) & \hat{u}(k-3) & \hat{u}(k-4) & 0 & 0 & 0 \\
0 & 0 & 0 & \hat{u}(k) & \hat{u}(k-1) & \hat{u}(k-4)
\end{bmatrix}
\]  \hspace{1cm} (6.15)

Similarly from Equation (3.23) one knows that Alternative 2 is given by

**Alternative 2: Weighted Residual Estimate of \( u(k) \)**

\[
\hat{u}(k) = P_1(k)[1] + P_0(k)[-1] = P_1(k) - P_0(k) \]  \hspace{1cm} (6.16)

where \( P_1(k), i\epsilon\{0,1\} \), is the a posteriori probability that
the \( i \)th model matches the actual system (Equation (3.7)).

\( P_i(k) \) is given by the following equation

\[
P_i(k) = \frac{p(z(k)/H_i, Z(k-1))}{\sum_{j=0}^{1} p(z(k)/H_j, Z(k-1))}
\]

(6.17)

where \( p(z(k)/H_i, Z(k-1)) \) is the Gaussian density which has already been defined in Equation (6.10).

Consequently once \( \hat{u}(k) \) has been evaluated, the observation matrix estimate \( \hat{C}(k) \) is straightforwardly constructed by suitably altering either \( C_1(k) \) or \( C_0(k) \) as follows:

\[
\hat{C}(k) = \begin{bmatrix}
\hat{u}(k) & \hat{u}(k-3) & \hat{u}(k-4) & 0 & 0 & 0 \\
0 & 0 & 0 & \hat{u}(k) & \hat{u}(k-1) & \hat{u}(k-4)
\end{bmatrix}
\]

(6.18)

Earlier in this chapter it was indicated that the initial state estimate \( \hat{x}(0/0) \) is assumed to be Gaussian with mean \( x_0 \) and covariance \( \Sigma(0/0) \). Moreover, in Chapter 1 (Equation (1.16)) the state vector \( x(k) \) was defined as

\[
x(k) = \begin{bmatrix}
1 \\
\rho R_s (\tau-(K-1)T) \\
\rho R_s (KT-\tau) \\
R_s (KT-\tau) \\
R_s (\tau-(K-1)T) \\
\rho
\end{bmatrix}
\]

(6.19)
where $R_s(\cdot)$ is the autocorrelation of the signal $s^*(\cdot)$

$\rho$ is the reflection coefficient with a value equally likely between 0 and 1

$\tau$ is the excess multipath delay

$T$ is the time between consecutive bit generation at the transmitter

$K$ equals $1 + \tau/T$

$k$ is the discrete time index

Since $\hat{x}(0/0)$ is the best a priori estimate one can make and $\rho$ is a uniform random variable between 0 and 1, then it is reasonable to set

$$
\hat{x}(0/0) = \begin{bmatrix}
1 \\
0 \\
0 \\
0 \\
.5
\end{bmatrix}
$$

Moreover, it can be shown that a reasonable initial covariance matrix is given by $[7^*]$
The adaptation of the overall Adaptive Joint Detection Estimation Algorithm to the Multipath Communication Case is now complete. In the next section computer simulation results for various state vector examples are studied.

6.3 Computer Simulation Results

In this section the computer simulation results of the Multipath Communication Case under both "high to moderate" and "low" signal to noise ratio conditions, using both Alternative 1 and Alternative 2 are presented.

The measurement sequence \( \{z(k)\} \) is generated by first using a random number generator with seeds to generate the transmitted bit sequence \( \{u(k)\} \) and a Gaussian distributed noise sequence \( \{v(k)\} \) with zero mean and unit variance. Then by selecting a value for the state \( x \) and implementing Equation (6.2), the measurement sequence \( \{z(k)\} \) is generated.

Earlier in this chapter and Chapter 5 it was pointed out that since the state dynamics is not driven by plant noise, an artificial Gaussian noise sequence is added to enhance
the performance of the algorithm. The Gaussian noise sequence used in all the computer simulations in this chapter has zero mean and covariance $Q(k)$ which is given by

$$Q = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 \\
0 & .01 & 0 & 0 & 0 & 0 \\
0 & 0 & .01 & 0 & 0 & 0 \\
0 & 0 & 0 & .01 & 0 & 0 \\
0 & 0 & 0 & 0 & .01 & 0 \\
0 & 0 & 0 & 0 & 0 & .01 \\
\end{bmatrix} \quad (6.22)$$

Table 6.1 summarizes all the computer simulation results discussed in this section. Note two distinct situations were examined: "high to moderate" SNR and "low" SNR.

"High to Moderate" Signal to Noise Ratio

Under these auspicious SNR conditions the input data bit estimate $\hat{u}(k)$ generated by the weighted residual approach (Alternative 2) is essentially the same as the one furnished by the maximum likelihood detector (Alternative 1). This is attributable to the fact that under "high" SNR conditions the a posteriori probabilities $P_1(k)$ and $P_0(k)$ are very close to 0 and 1 or vice versa. Intuitively this implies that one can determine almost unequivocally which is the actual model (hypothesis) since the received signal is barely corrupted with noise. Therefore, since under Alternative 1 only the
<table>
<thead>
<tr>
<th>DESCRIPTION</th>
<th>( \hat{x}_1 )</th>
<th>( \hat{x}_2 )</th>
<th>( \hat{x}_3 )</th>
<th>( \hat{x}_4 )</th>
<th>( \hat{x}_5 )</th>
<th>( \hat{x}_6 )</th>
<th>RMS ERROR</th>
<th>( \hat{x}_1 )</th>
<th>( \hat{x}_2 )</th>
<th>( \hat{x}_3 )</th>
<th>( \hat{x}_4 )</th>
<th>( \hat{x}_5 )</th>
<th>( \hat{x}_6 )</th>
<th>NUMBER OF DETECTION ERRORS</th>
<th>FIGURES</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;HIGH-TO MODERATE&quot; SNR</td>
<td>( x = [1.0 \ 0. \ 9 \ 1.0 \ 0. \ 9]^T )</td>
<td>1.00</td>
<td>.015</td>
<td>.778</td>
<td>.989</td>
<td>.058</td>
<td>.933</td>
<td>0.00</td>
<td>.185</td>
<td>.288</td>
<td>.205</td>
<td>.159</td>
<td>.137</td>
<td>0</td>
<td>6.1(a), (b), (c)</td>
</tr>
<tr>
<td>( E = 4.0 )</td>
<td>( x = [1.0 \ 0. \ 9 \ 1.0 \ 0. \ 9]^T )</td>
<td>1.00</td>
<td>-.075</td>
<td>.843</td>
<td>.999</td>
<td>.125</td>
<td>.904</td>
<td>0.00</td>
<td>.245</td>
<td>.377</td>
<td>.322</td>
<td>.217</td>
<td>.152</td>
<td>0</td>
<td>6.2(a), (b), (c)</td>
</tr>
<tr>
<td>&quot;LOW&quot; SNR</td>
<td>( x = [1.0 \ 0. \ 9 \ 1.0 \ 0. \ 9]^T )</td>
<td>1.00</td>
<td>-.036</td>
<td>.691</td>
<td>.737</td>
<td>-.311</td>
<td>.214</td>
<td>0.00</td>
<td>.116</td>
<td>.695</td>
<td>.673</td>
<td>.444</td>
<td>.594</td>
<td>0</td>
<td>6.3(a), (b), (c)</td>
</tr>
<tr>
<td>( E = 1.0 )</td>
<td>( x = [1.0 \ 0. \ 9 \ 1.0 \ 0. \ 9]^T )</td>
<td>1.00</td>
<td>.008</td>
<td>.930</td>
<td>1.008</td>
<td>-.221</td>
<td>.282</td>
<td>0.00</td>
<td>.113</td>
<td>.488</td>
<td>.506</td>
<td>.314</td>
<td>.385</td>
<td>11</td>
<td>6.4(a), (b), (c)</td>
</tr>
<tr>
<td>( E = 2.0 )</td>
<td>( x = [1.0 \ 0. \ 9 \ 1.0 \ 0. \ 9]^T )</td>
<td>1.00</td>
<td>.463</td>
<td>.552</td>
<td>1.24</td>
<td>-.373</td>
<td>1.014</td>
<td>0.00</td>
<td>.292</td>
<td>.386</td>
<td>.312</td>
<td>.189</td>
<td>.220</td>
<td>6</td>
<td>---</td>
</tr>
<tr>
<td>( E = 2.0 )</td>
<td>( x = [1.0 \ 0. \ 9 \ 1.0 \ 0. \ 9]^T )</td>
<td>1.00</td>
<td>-.116</td>
<td>.450</td>
<td>1.145</td>
<td>-.499</td>
<td>.697</td>
<td>0.00</td>
<td>.274</td>
<td>.450</td>
<td>.273</td>
<td>.239</td>
<td>.219</td>
<td>7</td>
<td>---</td>
</tr>
</tbody>
</table>

**TABLE 6.1**

COMPUTER SIMULATION RESULTS OF MULTIPATH COMMUNICATION CASE
maximum likelihood hypothesis is selected, then the data bit sequence estimate \( \{\hat{u}(k)\} \) generated by both alternatives of the AJDEA are for the most part identical. Consequently, the state estimate \( \hat{x}(k/k) \) generated by both alternatives are approximately equal. Extensive computer simulations were performed that attest to this assertion.

For example, Figure 6.1 is a relatively "high" SNR case. The unknown state vector is \([1.0 \ 0 \ 0.9 \ 1.0 \ 0.9]^T\). Note in particular that the reflection coefficient is .9. As expected the state estimate \( \hat{x}(k/k) \) quickly locks on the correct values with very low RMS errors. After fifty iterations \( \hat{x}(k/k) \) is \([1.00 \ 0.015 \ 0.778 \ 0.989 \ 0.054 \ 0.933]^T\) with an RMS error over fifty iterations of \([0.0 \ \ 0.185 \ 0.288 \ 0.205 \ 0.159 \ 0.137]^T\). Moreover, the unknown data bit sequence estimate \( \{\hat{u}(k)\} \) is excellent with no detection errors made in identifying forty-nine transmitted data bits.

The major objective of this thesis has been attained, at least for high SNR conditions of the Multipath Communication Case. In particular an excellent state estimate and a perfectly identified data sequence were generated.

Extensive computer simulations for a wide spectrum of SNR conditions were conducted. However, only the few exemplary ones listed in Table 6.1 will be discussed. The next step is to see if the AJDEA performs satisfactorily under lower SNR conditions. In Figure 6.2 the results from
Fig. 6.1(a)

\[ E = 14.0 \]
Fig. 6.1(b)

$E = 14.0$
Fig. 6.1(c)

$E = 14.0$
an intermediate case are presented. Note that the signal energy $E$ equals 4, which is substantially lower than in Figure 6.1 in which $E$ equals 14. Moreover, as a reminder, the measurement noise $v(k)$ was defined as zero mean with unit variance. However, as illustrated in Figure 6.2 and Table 6.1, the algorithm still performs well. The state vector is $[1.0 \ 0. \ .0 \ 1.0 \ .0 \ .9]^T$. The state estimate $\hat{x}(k/k)$ locks on the correct values with low RMS errors. In particular after fifty iterations $\hat{x}(k/k)$ is $[1.00 \ -.075 \ .843 \ .999 \ .125 \ .904]^T$ with an RMS error over fifty iterations of $[0.00 \ .245 \ .377 \ .322 \ .217 \ .152]^T$. In addition, the unknown data bit sequence estimate $\{u(k)\}$ is excellent with no detection errors made in identifying forty-nine transmitted data bits.

"Low" Signal to Noise Ratio

As anticipated, the Adaptive Joint Detection Estimation Algorithm does not generally perform as well under "low" SNR conditions as it did under "high to moderate" SNR conditions. The number of detection errors and the estimation errors can generally be expected to increase. In this section, as delineated in Table 6.1, computer simulation results implementing both Alternatives 1 and 2 are presented and compared. In the preceding two chapters it was shown that Alternative 1 performed much better than Alternative 2 for the Scalar Case. However, this significant edge did not carry over to
Fig. 6.2(a)

$E = 4.0$
Fig. 6.2(c)

E = 4.0
the Two-Dimensional Vector Case. The overall performance of Alternative 1 proved to be only slightly superior to that of Alternative 2. The simulation results of this section suggest a surprising property. The performance of Alternative 2 is at least commensurate and often superior to that of Alternative 1 for the Multipath Communication Case. This can be seen by studying the results presented in the bottom four rows of Table 6.1 and the associated figures.

First consider Figure 6.3, which represents the very "low" SNR case where the signal energy $E$ equals 1. Alternative 1 was used to generate the state and data bit estimates. The unknown state vector is $[1.0 \ 0.0 \ 0.9 \ 1.0 \ 0.0 \ 0.9]^T$. Obviously different measurement and data bit sequences from the ones used in Figures 6.1 and 6.2 were used in the computer simulation. Unlike for higher SNR conditions, the state estimate $\hat{x}(k/k)$ does not generally lock on the true values. After fifty iterations the state estimate $\hat{x}(k/k)$ is $[1.0 \ -0.036 \ 0.691 \ 0.737 \ -0.311 \ 0.214]^T$ with an RMS error over fifty iterations of $[0.0 \ 0.116 \ 0.695 \ 0.673 \ 0.444 \ 0.507]^T$. Clearly, the final state estimate $\hat{x}(50/50)$ is not satisfactory and the RMS errors are much higher than those in Figures 6.1 and 6.2. Moreover, the unknown data bit sequence estimate $\{\hat{u}(k)\}$ is disappointing with eleven detection errors made in identifying forty-nine transmitted data bits.

Therefore, for the first time we have encountered a case
Fig. 6.3(a)
Alternative 1
$E = 1.0$
Fig. 6.3(b)
Alternative 1
$E = 1.0$
Fig. 6.3(c)
Alternative 1
$E = 1.0$
where the performance of AJDEA - Alternative 1 is unsatisfactory. Previously, in the Scalar Case and the Two-Dimensional Vector Case for very "low" SNR conditions the state estimate \( \hat{x}(50/50) \) is sufficiently "good" to warrant the use of the optimal Viterbi algorithm to generate a data sequence estimate with a lower number of detection errors. However, in this case the data sequence estimate cannot be improved by employing the Viterbi algorithm since the state estimate \( \hat{x}(50/50) \) is so poor.

The next step is to apply the weighted residual approach (Alternative 2) to the same case as we considered in Figure 6.3, in order to determine if it furnishes a better state estimate. The results are presented in row 4 of Table 6.1 and Figure 6.4.

The unknown state vector is \([1.0 \ 0. \ .9 \ 1.0 \ 0. \ .9]^T\). Moreover, the same measurement \(\{z(k)\}\) and data sequence \(\{u(k)\}\) as in Figure 6.3 are used in the computer simulations. Studying Figure 6.4 one can see that unlike the preceding figure the state estimate \( \hat{x}(k/k) \) generally locks on the true values. After fifty iterations the state estimate \( \hat{x}(50/50) \) is equal to \([1.0 \ .008 \ .980 \ 1.008 \ -.221 \ .282]^T\) with an RMS error over fifty iterations of \([0.0 \ .113 \ .488 \ .506 \ .314 \ .385]^T\). Note by comparing to the RMS error values in row 3 of Table 6.1 that there is an across-the-board decrease in RMS error. In fact, except for the sixth element \( \hat{x}_6(50/50) \), the final
Fig. 6.4(a)
Alternative 2
E = 1.0
Fig. 6.4(b)
Alternative 2
$E = 1.0$
Fig. 6.4(c)
Alternative 2
E = 1.0
state estimate \( \hat{x}(50/50) \) is excellent considering the poor SNR conditions. However, once again the unknown data bit sequence estimate \{u(k)\} is disappointing with eleven detection errors made in identifying forty-nine transmitted data bits. But in this case the state estimate \( \hat{x}(50/50) \) is sufficiently good to warrant the implementation of the Viterbi algorithm to perform the "optimal" identification of the transmitted data sequence.

Various other simulations were performed for "low" SNR conditions with similar results. Generally the performance of the weighted residual approach (Alternative 2) is superior to that of maximum likelihood detection (Alternative 1). The closest situation that was encountered is delineated in rows 5 and 6. The choice between the two alternatives is a toss-up.

Therefore, based on all the simulation results, we conclude that the AJDEA performs satisfactorily for "low" signal to noise ratio. Moreover, in general Alternative 2 provided a better overall performance than Alternative 1.

6.4 Special Cases with Step Changes of the State Vector

In this section motivation is first provided for studying Multipath Communication Cases that experience unexpected step changes in the state vector. The next step is to adjust the dynamic system Equations (6.1) and (6.2) to encompass a
larger selection of multipath systems; in particular to include systems that experience sudden step changes of the state vector.

The motivation for introducing step changes of the state vector is to further test the robustness of the Adaptive Joint Detection Estimation Algorithm. The algorithm should be able to recover from sudden step changes of any one of the elements of the state vector and quickly track the new values. It was primarily to ensure an adaptive algorithm that we insisted earlier on the addition of artificial white noise for maintaining the magnitude of the gains of the estimator. If the gain magnitudes are allowed to decrease too rapidly, the estimator eventually runs open loop and ignores future measurements. Clearly, under such conditions the estimator cannot rapidly react to the sudden step changes. (For further discussion the reader is referred to Section 5.3).

The next step is the actual modification of the dynamic state Equations (6.1) and (6.2) to account for the sudden step changes that may occur. This situation can easily be incorporated into the system model by adding an extra term to the state dynamics equations:

\[
\text{State Dynamics}
\]

\[
x(k+1) = x(k) + a_δ(k,j)
\]  \hspace{1cm} (6.23)
where $\mathbf{a}$ is an unknown \((6 \times 1)\) vector which represents the change in magnitude of the elements of the state vector \(x(k)\). Clearly, not all the elements of $\mathbf{a}$ have to be nonzero. $\delta(k,j)$ is the Kroenecker delta function given by

\[
\delta(k,j) = \begin{cases} 
1 & \text{if } k=j \\
0 & \text{if } k \neq j 
\end{cases}
\]  

(6.24)

where $j \in \{1, 2, 3, \ldots\}$ is the unknown time increment which denotes when the step change occurs.

However, the measurement Equation (6.2) remains unaffected, i.e.,

Measurement Equation

\[
z(k) = C(k)x(k) + v(k)
\]  

(6.25)

By allowing unexpected step changes of the state vector the total uncertainty of the system has increased. It is not known when during the duration of the data transmission the step change occurred or if it occurred at all. If a step change does occur the magnitude is unknown.

It was indicated earlier that these step changes are due to sudden changes in the reflection coefficient - a situation that may often be encountered in communication systems.

Computer simulations were performed that involved sudden step changes of the reflection coefficient $\rho$. The results
demonstrate that in spite of the added uncertainty the AJDEA still performs well. The detector still generates an excellent data bit sequence estimate \( \hat{u}(k) \) with low detection errors. Meanwhile, the estimator furnishes a state vector estimate \( \hat{x}(k/k) \) that successfully tracks step jumps of the elements of the state vector. However, for brevity the results of only one exemplary case are presented and discussed in Section 6.5.

6.5 Further Computer Simulation Results

The case considered in this section falls into the "high to moderate" SNR class since the signal energy level is high (34) and the corruptive measurement noise used in the simulation had unit variance. Furthermore, it was established in Section 6.2 that for such systems the results furnished by either alternative are for the most part identical. Therefore, only the simulation results obtained using Alternative 2 (weighted residual estimate) are presented in Figure 6.5(a), (b), and (c).

The unknown state vector is once again \( x = \left[ \begin{array}{cccc} 1.0 & 0.0 & .9 & 1.0 & 0.0 & .9 \end{array} \right]^T \) up to the time \( k=25 \). Note in particular that the sixth element, the reflection coefficient \( \rho \), is equal to (.9). However, at \( k=25 \) the reflection coefficient jumps down to and remains thereafter at the value (.5). Hence, the state vector after \( k=25 \) is \( \left[ \begin{array}{cccc} 1.0 & 0.0 & .5 & 1.0 & 0.0 \end{array} \right] \).
Fig. 6.5(a)
Step Change in the Reflection Coefficient from $\rho = .9$ to $\rho = .5$ at $k = 25$
Fig. 6.5(b)
Step Change in the Reflection Coefficient
from $\rho = 0.9$ to $\rho = 0.5$ at $k = 25$
Fig. 6.5(c)
Step Change in the Reflection Coefficient from $\rho = .9$ to $\rho = .5$ at $k = 25$
Clearly, it is imperative that the estimator reacts as quickly as possible to the sudden change that takes place at \( k=25 \) to the stream of residuals that it receives. In other words, it is important that the gain \( G(k) \) be sufficiently large when the step jump occurs. This can be assured by carefully selecting a large enough artificial process noise covariance \( Q \). It has been emphasized that engineering judgment must be employed in selecting the value of \( Q \). In general, the larger the noise covariance, the larger the average RMS errors of the state estimate. Thus, there is, in a sense, a tradeoff between "responsiveness" and average RMS error.

Now returning back to Figure 6.5 we see that after twenty-four iterations the state vector estimate is \( [1.00 \ -0.039 \ 0.999 \ 0.865 \ 0.126 \ 0.988]^T \) and has locked onto the correct values. Immediately at \( k=25 \) the estimate begins tracking the step changes and by \( k=36 \) has locked onto the new values of the state vector. In particular after fifty iterations the state vector estimate is \( [1.00 \ 0.031 \ 0.287 \ 0.937 \ 0.023 \ 0.438]^T \). Most important of all the step change has not adversely affected the performance of the detector of the AJDEA. In fact the unknown data bit sequence estimate \( \hat{u}(k) \) is perfect with no detection error made in identifying forty-nine unknown transmitted bits.
6.5 Summary

In this chapter a number of key issues of the Adaptive Joint Detection Estimation Algorithm were studied by carefully examining numerous exemplary simulation results of the Multipath Communication Case.

Earlier it was emphasized that the major objective in this thesis was to formulate an adaptive and high-throughput procedure for simultaneously generating the state estimate and detecting the transmitted data bit sequence. The results have shown that the AJDEA is such a procedure for the Multipath Communication Case. Furthermore, Alternative 2 (weighted residual estimation) is expected to provide a better state estimate than Alternative 1 (maximum likelihood detection). This edge is most particularly evident for "low" SNR conditions.

One issue that did not receive any attention thus far is the necessity of transmitting the initialization bit. This issue was discussed at great length in the previous two chapters, in particular Chapter 4. In order to avoid unnecessary repetition the reader is referred to Section 4.2. Therefore, we will only state that to ensure proper operation of the algorithm an initialization bit should be transmitted.

Hence, the throughput of the algorithm which was defined in Equation (1.19) is
Throughput = \frac{\Delta}{1 + N} \quad (6.26)

where \( N \) is the length of the actual data transmission. Clearly, for reasonable values of \( N \) the algorithm provides high-throughput. Furthermore, in Section 6.4 the scope of the Multipath Communication Case is extended to include systems that can experience sudden step changes of the state vector. Simulation results show that the algorithm continues to operate satisfactorily and thus manifest the robustness of the Adaptive Joint Detection Estimation Algorithm.
CHAPTER 7

COMMENTS AND CONCLUSIONS

This thesis report has presented an analysis of the behavior of a specific class of communications systems that are dependent on unknown time-varying parameters. In particular, the motivation for this thesis originated from the communication problem investigated by Schneider and McGarty in "Reliable Satellite Communications for a Specular Multipath Channel" [29]. They propose a suboptimal scheme for situations in mobile satellite communications systems when complete knowledge of all the multipath parameters (in particular the reflection coefficient) is not readily available.

Although it is assumed that the source generates independent bits, the multipath channel introduces memory to the system. Forney proposed the conversion of a channel with memory to an equivalent system in which the original data source drives a finite state machine whose contents are weighted by suitable tap gains. Next, these weighted bits are summed and transmitted over a memoryless channel which is corrupted by additive white Gaussian noise.

Schneider and McGarty propose estimating the sequence of shift registers (i.e., input binary sequence) by processing the direct and specular multipath returns with the Viterbi
algorithm (metric computer). However, before the algorithm can be implemented, the channel tap gains must be determined. A known "learning" sequence is transmitted in order to generate the optimal Kalman filter state vector estimate.

But, by transmitting a learning sequence, one is sacrificing system throughput. This loss can be substantial for systems that are subject to fluctuations of channel characteristics. The major objective of this thesis is to formulate an efficient, high-throughput, adaptive detection-estimation algorithm. The algorithm should be capable of constantly monitoring the channel, so as to be able to quickly detect and track sudden changes in elements of the state vector (i.e., modifications to the tap gains of the finite state machine). It has been assumed that the system is reasonably well-behaved, in that changes are reasonably interspersed in time.

Thus far, a brief review of the motivation and goals of this thesis has been provided. The purpose of the remainder of this chapter is threefold. First, in Section 7.1, a modification is proposed for the update error covariance equation of Alternative 2. The modified update equation should better reflect the true behavior of the estimator and enhance the overall performance of the algorithm. The second purpose is to provide an overview of the major con-
clusions of the thesis, which is done in Section 7.2. Finally, in Section 7.3, a list of suggestions for future research is provided.

7.1 Modifications

This section is devoted to a brief discussion of a proposed modification to the update error covariance equation of Alternative 2. The formulation is similar to that of Bar-Shalom and Tse [5] in the derivation of a "weighted residual" multitarget tracking filter, which they called the "Probabilistic Data Association" filter.

The AJDEA prediction and update equations were derived in Appendix B. Various intermediate equations that were used in the derivation of the update error covariance equations in Appendix B will be necessary for this formulation. Therefore, a brief review of these items is given below. However, it is suggested that the reader review pertinent sections of Appendix B.

The dynamic system is defined by Equations (B.1) and (B.2) to be

\[
\begin{align*}
\dot{x}(k+1) &= x(k) + w(k) \\
\bar{z}(k) &= C(k)x(k) + v(k)
\end{align*}
\] (7.1)

It was shown that a growing bank of Kalman filters is needed to generate the optimal state estimate \( \hat{x}(k/k) \), since \( C(k) \) contains the unknown time-varying input bit \( u(k) \). However, it was pointed out that the growing bank can be delimited to
two parallel filters corresponding respectively to hypotheses
$H_1$ (i.e., $u(k)=1$) and $H_0$ (i.e., $u(k)=-1$). The elemental
state estimates generated by these two parallel filters are given
by Equations (B.9) and (B.10), namely

$$\hat{x}_1(k/k) = \hat{x}(k-1/k-1) + K_1(k)r_1(k) \quad (7.3)$$
$$\hat{x}_0(k/k) = \hat{x}(k-1/k-1) + K_0(k)r_0(k) \quad (7.4)$$

where $K_1(k)$ and $K_0(k)$ are the filter gains corresponding
to hypotheses $H_1$ and $H_0$ respectively.

The crux of Alternative 2 is to use the weighted
residual $r(k) = P_1(k)r_1(k) + P_0(k)r_0(k)$, where $P_1(k)$ and
$P_0(k)$ are the a posteriori probabilities that $H_1$ and $H_0$
are true respectively.

Therefore, from Equations (B.13) and (B.14),

$$\hat{x}_1(k/k) = \hat{x}(k-1/k-1) + K_1(k)r(k) \quad (7.5)$$
$$\hat{x}_0(k/k) = \hat{x}(k-1/k-1) + K_0(k)r(k). \quad (7.6)$$

Now define the error of the state estimate of the Kalman
filter conditioned on $H_1$ to be $\tilde{x}_1(k/k)$, i.e.,

$$\tilde{x}_1(k/k) \triangleq x(k) - \hat{x}_1(k/k). \quad (7.7)$$

Similarly define

$$\tilde{x}_0(k/k) \triangleq x(k) - \hat{x}_0(k/k) \quad (7.8)$$
and the overall (unconditioned) state error to be

\[ \tilde{x}(k/k) \triangleq x(k) - \hat{x}(k/k). \] (7.9)

Moreover, define the conditional state estimate error covariance \( \Sigma_1(k/k) \) to be

\[ \Sigma_1(k/k) \triangleq \mathbb{E}\{ \tilde{x}_1(k/k)\tilde{x}_1^T(k/k) \}. \] (7.10)

Similarly define

\[ \Sigma_0(k/k) \triangleq \mathbb{E}\{ \tilde{x}_0(k/k)\tilde{x}_0^T(k/k) \} \] (7.11)

and the overall state estimate error covariance to be

\[ \Sigma(k/k) \triangleq \mathbb{E}\{ \tilde{x}(k/k)\tilde{x}^T(k/k) \}. \] (7.12)

In Appendix B it was assumed that the overall state estimation error \( \tilde{e}(k) \) is zero mean during the derivation of the error covariance update Equation (B.31). In this chapter this assumption is relaxed and a modified error covariance update equation is obtained.

Equation (7.12) can be easily rewritten as

\[ \Sigma(k/k) = \int [\tilde{x}(k/k)\tilde{x}^T(k/k)] \mathbb{E} [\tilde{x}(k/k)\tilde{x}^T(k/k)] dx \] (7.13)

where \( \mathbb{E} [\tilde{x}(k/k)\tilde{x}^T(k/k)] \) is the conditional a posteriori density and \( Z(k) = \{ z(1), z(2), ..., z(k) \} \).

The two hypotheses \( H_1 \) and \( H_0 \) are mutually exclusive and collectively exhaustive, and thus constitute an event.
By using Bayes' Rule the density in Equation (7.13) can be expressed as

\[
p[x(k)/Z(k)] = \sum_{i=0}^{1} p[x(k)/H_i, Z(k)]P[H_i/Z(k)]
\]  

(7.14)

where \(P_i(k) = P[H_i/Z(k)]\) is the a posteriori probability that \(H_i\) is the true hypothesis. Substituting Equation (7.14) into (7.13) yields

\[
\Sigma(k/k) = \int [\hat{x}(k/k)\hat{x}^T(k/k)] \sum_{i=0}^{1} p[x(k)/H_i, Z(k)]P_i(k)dx
\]

= \sum_{i=0}^{1} P_i(k)E{(x(k)-\hat{x}(k/k))(x(k)-\hat{x}(k/k))^T}.

(7.15)

After considerable algebraic manipulation (which is presented in Appendix D), it can be shown that

\[
\Sigma(k/k) = [I-G(k)\hat{C}(k)]\Sigma(k/k-1) + \sum_{i=0}^{1} P_i(k)K_i(k)r(k)r^T(k)K_i^T(k)
\]

- \(G(k)r(k)r^T(k)G^T(k)\)

(7.16)

where \(r(k) = z(k) - \hat{C}(k)\hat{x}(k/k-1)\). The gains \(K_1(k)\) and \(K_0(k)\) are updated at each iteration, i.e.,

\[
K_1(k) = \Sigma(k/k-1)C_1^T(k)[C_1(k)\Sigma(k/k-1)C_1^T(k) + R(k)]^{-1}
\]  

(7.17)
where \( \epsilon(0, 1) \) and \( R(k) \) is the "inflated" measurement noise covariance matrix which is discussed in Appendix B.

Let

\[
\Sigma^*(k/k) = [I - G(k)\hat{X}(k)]\Sigma(k/k-1)
\]

(7.18)

and

\[
\Sigma^0(k/k) = \sum_{i=0}^{1} P_1(k)K_1(k)r(k)r^T(k)K_1^T(k) - G(k)r(k)r^T(k)G^T(k).
\]

(7.19)

Therefore, Equation (7.16) can be rewritten as

\[
\Sigma(k/k) = \Sigma^*(k/k) + \Sigma^0(k/k).
\]

Note that, as expected \( \Sigma^0(k/k) \) is zero when either \( P_1(k) \) or \( P_0(k) \) equals 1. It can be shown, using the Schwartz inequality [5] that \( \Sigma^0(k/k) \) is a positive semidefinite matrix. It is felt that the modified covariance \( \Sigma(k/k) \) is a more realistic characterization of the uncertainty in the estimator output. It remains to be shown, based on numerical simulation, that the addition of the extra term \( \Sigma^0(k/k) \) enhances the performance of Alternative 2 of the AJDEA.

7.2 Summary of Conclusions

The major conclusions and results of this thesis are of two basic classes. The first class consists of specific results which pertain directly to the specific cases studied
in particular the Multipath Communication Case. The second class is comprised of those general conclusions and results that are interspersed throughout this thesis report. Generally, these are the consequences of extrapolating the specific results of the first class. Both classes of conclusions and results are summarized below.

7.2.1 Specific Conclusions

A new joint detection estimation algorithm was formulated for efficiently detecting the unknown data bit sequence and for estimating the state vector from the received measurements. Computer simulation results demonstrate that effective channel equalization has been attained with this technique. Moreover, this technique is more adaptive and provides higher throughput capability than those techniques proposed in Godard [8], and Schneider and McGarty [29]. Listed below are the major characteristics of the algorithm:

a) There is strong mutual coupling between the estimator and detector of the AJDEA. The schedule of operation is a bit-by-bit detect-then-estimate protocol.

b) The AJDEA is a high-throughput algorithm that generally only requires one initialization bit.

c) The structure of the AJDEA is such that it constantly
monitors the system and tracks sudden changes. In Chapters 5 and 6 the algorithm successfully tracked step changes in various elements of the state vector with no major performance degradation for "reasonable" signal to noise ratio conditions. In the latter chapter this sudden step change was attributed to sudden changes of the reflection coefficient.

d) Two alternative detectors were proposed and implemented in this thesis. Alternative 1 consists mainly of a standard maximum likelihood detection test. Only the hypothesis (model) with the larger a posteriori probability $P_i(k)$ is retained. Thus, only one Kalman filter must actually be implemented. Alternative 2 is based on a distinctly different approach. The alternative entails using the weighted residual instead of the elemental residuals. Computer simulations of the three specific cases were performed using both alternatives with the following conclusions. It was demonstrated in Chapter 4 that Alternative 1 performed better than Alternative 2 for the Scalar Case examples considered. The results in Chapter 5 showed that Alternative 1 performed only slightly better than Alternative 2 for the Two-
Dimensional Vector Case examples simulated. On the other hand, the Multipath Communication Case simulations provided the surprising results that Alternative 2 generally performed better than Alternative 1. In particular, the number of detection errors was equal, but Alternative 2 state estimates generally had smaller RMS errors than Alternative 1. An analogous property was obtained by Athans et al. in "A Suboptimal Estimation Algorithm with Probabilistic Editing for False Measurements with Applications to Target Tracking with Wake Phenomena" [1]. Athans et al. derive a probabilistic edit algorithm that is based upon two distinct hypotheses corresponding respectively to the measurement being true or false. This edit algorithm uses not only the prior probabilities of a simple $X^2$-edit, but also the a posteriori probabilities computed in the Kalman filter update cycle.

7.2.2 General Conclusions

The general conclusions listed below were reached by extending various specific conclusions.

The weighted residual technique is a promising approach for processing sensor measurements of stochastic systems that are dependent on unknown time-varying parameters.
The staggered detect-then-estimate protocol forms strong mutual coupling between the detector and the estimator of the system processor. This coupling provides the processor with the capability of monitoring the system for sudden unexpected changes. For example, stochastic systems that are prone to sensor failures should be monitored to detect such failures.

7.3 Suggestions for Future Research

Suggestions for future research endeavors fall into two classes: those addressing specific characteristics of the multipath communication problem and the Adaptive Joint Detection Estimation Algorithm, and those on a more general, conceptual level.

7.3.1 Specific

a) In Section 7.1, a modification to the error covariance \( \Sigma(k/k) \) update equation of Alternative 2 is proposed. Computer simulations of the three cases (defined in Chapter 2) employing the modified Alternative 2 estimator are needed to determine whether the modification enhances the performance of the algorithm.

b) An analytic stability study is necessary in order to make the formulation of the algorithm complete. The computer simulation results in this thesis
have shown that the algorithm performed well for the limited number of cases considered. However, no performance bounds have as yet been evaluated for the AJDEA. It is important to analytically evaluate the effects of measurement uncertainty on the performance of the decoder and in turn, on the consistency of the state estimate. This measurement uncertainty was accounted for in Appendix B by effectively "inflating" the measurement noise. Hence, bounds for the probability of detection error and state estimation biases that are a function of the signal to "inflated" measurement noise need to be determined.

c) In the derivation of the AJDEA in Chapter 3 and Appendix B, the growing bank of Kalman filters are originally delimited to a bank of two Kalman filters, which are conditioned on the two incremental models (hypotheses H₁ and H₀) respectively. The next step was to use either the weighted residual approach or maximum likelihood detection to further collapse the bank of two filters to one "pseudo" Kalman filter. An alternative technique is to stop at the intermediate step and use the bank of two Kalman filters. In particular, the elemental state estimates
\( \hat{x}_1(k/k) \) and \( \hat{x}_0(k/k) \) and their respective filter gains, \( K_1(k) \) and \( K_0(k) \), would have to be calculated at each iteration. Ricker and Williams [37] use such a technique for tracking maneuvering targets.

d) The state space representation of the multipath communication problem studied in this thesis was shown, in Chapter 1, to be a function of two multipath parameters: the reflection coefficient \( \rho \) and the multipath delay \( \tau \). It was assumed that the delay \( \tau \) is "known," while the reflection coefficient \( \rho \) must be estimated. The responsiveness of the algorithm was tested by introducing sudden step changes in the value of \( \rho \). This test only constituted a rudimentary sensitivity analysis of the algorithm. More sophisticated tests should be conducted to ascertain the sensitivity of the algorithm to a large variety of multipath parameter changes.

e) This suggestion follows directly from the previous one. Both in McGarty and Schneider [29] and in this thesis, the assumption was made that one has direct knowledge of the timing of the multipath return (i.e., the delay \( \tau \)). However, there are situations in which the value of this parameter is
not readily available. For these situations a totally new detection-estimation algorithm would have to be formulated.

7.3.2 General

In the derivation of the AJDEA - Alternative 2 we successfully delimited the growing bank of Kalman filters to a bank of two filters, which are conditioned on the two incremental models (hypotheses) respectively. Furthermore, by driving these filters with the weighted residual, the bank of filters is collapsed to a single "pseudo" Kalman filter. A significant endeavor would be a detailed study of the weighted residual technique, highlighted with applications to other stochastic systems that are dependent on unknown time-varying parameters.

The general theory of detection and estimation has applications in many diverse classes of problems. Separability is a major issue that is invariably encountered when analyzing such problems. The issue that must be resolved is whether mutual coupling between the detection and estimation procedures or completely separate operation is desirable. (In this thesis we chose to follow a bit-by-bit detect-then-estimate protocol, which entails strong mutual coupling). It remains to be determined how to resolve the separability issue for any particular detection-estimation problem.
APPENDIX A

REVIEW OF THE MULTIPLE MODEL ESTIMATION ALGORITHM AND THE VITERBI ALGORITHM

In this Appendix brief descriptions of the Multiple Model Estimation Algorithm and the Viterbi algorithm are presented.

A.1 Review of the Multiple Model Estimation Algorithm

The purpose of this section is to introduce the Multiple Model Estimation Algorithm (MMEA). A full description will not be given as it is available in many other sources [4, 8*, 20, 22].

Consider a linear stochastic dynamical system whose dynamics are a function of a parameter $\theta$. The system can be expressed by the following pair of discrete state space equations:

**State Dynamics**

$$x(k+1) = A(\theta)x(k) + B(\theta)y(k) + L(\theta)x(k). \quad (A.1)$$

**Measurement Equation**

$$z(k) = C(\theta)x(k) + v(k). \quad (A.2)$$

Although not explicitly shown $A$, $B$, $C$, and $L$ may be time-varying. $x(k)$ is the $n$-dimensional state vector, $y(k)$ is
the m-dimensional input vector and \( z(k) \) is the p-dimensional measurement vector. The noise sources \( w(k) \) and \( v(k) \) are assumed to be zero mean white Gaussian noises of covariance \( Q(k) \delta(k,j) \) and \( R(k) \delta(k,j) \) respectively, where \( \delta(k,j) = \begin{cases} 1 & k=j \\ 0 & k \neq j \end{cases} \). The matrices \( A, B \) and \( C \) are the (nxn) system, (nxm) input and (pxn) observation matrices respectively. The following notation \((A, B, C, L)\) will be used when referring to the linear system given by Equations (A.1) and (A.2).

However, the actual value of \( A, B, C, \) or \( L \) may not be known since they are dependent on an unknown parameter \( \theta \). Let \( M \) be the number of possible values of parameter \( \theta \). Therefore, there are \( M \) "hypothetical" models of the actual system, where the \( i \)th model is represented by \((A(\theta_i), B(\theta_i), C(\theta_i), L(\theta_i))\) and is given by:

\[
x(k+1) = A(\theta_i)x(k) + B(\theta_i)p(k) + L(\theta_i)w(k). \tag{A.3}
\]

\[
z(k) = C(\theta_i)x(k) + v(k). \tag{A.4}
\]

For simpler notational purposes, let \( A_1 = A(\theta_1), B_1 = B(\theta_1), C_1 = C(\theta_1), \) and \( L_1 = L(\theta_1) \). Thus, the \( i \)th model represented by \((A_1, B_1, C_1, L_1)\) can be expressed as:

\[
x(k+1) = A_1x(k) + B_1p(k) + L_1w(k). \tag{A.5}
\]

\[
z(k) = C_1x(k) + v(k). \tag{A.6}
\]
At this point it will prove useful to rewrite the following definitions found in Greene [8*]:

**Definition 1:** the $i$th Kalman filter is defined to be **matched** if the matrices used in the filter design (i.e., the model) and the matrices of the actual system are the same; i.e., if $A_i = A$, $B_i = B$, $C_i = C$, and $L_i = L$.

**Definition 2:** the $i$th Kalman filter is said to be **mismatched** if it is not matched.

For the $i$th model of the system the state estimate is generated by a matching Kalman filter defined by the following equations:

**Predict Cycle**

\[
\hat{x}_i(k+1/k) = A_i \hat{x}_i(k/k) + B_i u(k) \tag{A.7}
\]

\[
\Sigma_i(k+1/k) = A_i \Sigma_i(k/k) A_i^T + L_i Q(k) L_i^T \tag{A.8}
\]

**Update Cycle**

\[
\hat{x}_i(k/k) = \hat{x}_i(k/k-1) + K_i(k) [z(k) - C_i \hat{x}(k/k-1)] \tag{A.9}
\]

\[
K_i(k) = \Sigma_i(k/k-1) C_i^T [C_i \Sigma_i(k/k-1) C_i^T + R(k)]^{-1} \tag{A.10}
\]

\[
\Sigma_i(k/k) = [I - K_i(k) C_i] \Sigma_i(k/k-1) \tag{A.11}
\]

where

\[
\hat{x}_i(k/k) = E\{\hat{x}(k)/\theta = \theta_i, Z(k)\} \tag{A.12}
\]

\[
\hat{x}_i(k+1/k) = E\{\hat{x}(k+1)/\theta = \theta_i, Z(k)\} \tag{A.13}
\]
\[ \Sigma_1(k/k) = \text{cov}\{x(k); x(k)/\theta=\theta_1, Z(k)\} \]  
(A.14)

\[ \Sigma_1(k+1/k) = \text{cov}\{x(k+1); x(k+1)/\theta=\theta_1, Z(k)\} \]  
(A.15)

\[ Z(k) = \{u(0), u(1), u(2), \ldots, u(k-1), x(1), \ldots, x(k)\} \]  
(A.16)

\[ \hat{x}(0/0) = (x_0, \Sigma(0/0)) \]  
(A.17)

The basic filter structure of the Multiple Model Estimation Algorithm consists of a bank of Kalman filters as is shown in Figure A.1. Each Kalman filter is matched to a corresponding value of the parameter \( \theta \). Thus, for an M-ary model system, the filter structure consists of a bank of M Kalman filters. The adaptive estimate of the system state vector is generated by summing the elemental conditional state estimates of each filter which are weighted by the posteriori model (hypothesis) probabilities.

The notation that will be used for the weighting probabilities is given by

\[ P_i(k) \triangleq \text{Prob}(\theta=\theta_i/Z(k)) \]  
(A.18)

and

\[ \sum_{i=1}^{M} P_i(k) = 1 \]  
(A.19)
Figure A.1  MULTIPLE MODEL ESTIMATION ALGORITHM*

*Modified figure extracted from Fig. 3.1 in Reference [3]
\( P_i(0) \) is the a priori probability that the \( i \)th model is the actual one. Thus, \( P_i(0) = \text{Prob}(\theta = \theta_i) \) and from (A.19)

\[
\sum_{i=1}^{M} P_i(0) = 1. \tag{A.20}
\]

Using Baye's Rule it can be easily shown that

\[
P_i(k+1) = \frac{P(z(k+1)/\theta = \theta_i, Z(k))}{\sum_{j=1}^{M} P_j(k)p(z(k+1)/\theta = \theta_j, Z(k))} P_i(k). \tag{A.21}
\]

Since \( v(k) \) is Gaussian the density \( p(z(k+1)/\theta = \theta_i, Z(k)) \) is Gaussian. Using the standard notation for a Gaussian density

\[
p(z(k)/\theta = \theta_i, Z(k)) \sim N(C_i(k+1)\hat{x}_i(k+1/k), S_i(k+1)) \tag{A.22}
\]

where

\[
S_i(k+1) = C_i(k+1)\Sigma_i(k+1/k)C_i^T(k+1) + R(k+1)
\]

Equation (A.21) can be simplified if a different notation is utilized. Let the residual (a \( p \)-dimensional) vector generated by each Kalman filter be given by

\[
\Xi_i(k+1) = z(k+1) - C_i(k+1)\hat{x}_i(k+1/k) \tag{A.23}
\]

i.e., the difference between the actual measurement and the predicted measurement. Also let \( \sigma_i(k+1) \) be the scalar
precomputable quantity defined by

\[ \beta_1(k+1) \propto \frac{1}{2\pi} \left[ \det S_1(k+1) \right]^{-1/2}. \]

Therefore, the dynamic evolution of the a priori probability in Equation (A.21) can now be expressed as

\[ P_1(k+1) = \frac{\beta_1(k+1) \exp\left\{ -\frac{1}{2} r_1(k+1) S_1^{-1}(k+1) r_1(k+1) \right\} \sum_{j=1}^{M} \beta_j(k+1) \exp\left\{ -\frac{1}{2} r_j(k+1) S_j^{-1}(k+1) r_j(k+1) \right\} }{P_1(k)}. \]  

(A.24)

Thus a truly recursive update equation has been obtained for the posteriori probabilities. A block diagram illustration of the Multiple Model Estimation Algorithm (Bank of Kalman filters) is presented in Figure A.1.

**State Vector Estimation**

The overall state vector estimate is the Bayesian weighted average (by the posteriori probabilities \( P_1(k) \)) of the elemental state estimates generated by each one of the M filters in the bank of Kalman filters.

This fact can be easily shown.

\[ \hat{x}(k/k) = E\{x(k)/z(k)\} \]  

(A.25)

\[ = \int \hat{x}(k) p_x(x(k)/z(k)) dx(k). \]
However \( p(x(k)/Z(k)) = \sum_{i=1}^{M} P_i(k)p(x(k)/\theta=\theta_i, Z(k)) \). (A.26)

Therefore

\[
\hat{x}(k/k) = \sum_{i=1}^{M} P_i(k) \int x(k)p(x(k)/\theta=\theta_i, Z(k))dx(k)
\]

(A.27)

\[
= \sum_{i=1}^{M} P_i(k) \hat{x}_i(k/k).
\]

The true conditional covariance matrix \( \Sigma(k/k) \) is defined as

\[
\Sigma(k/k) \triangleq \text{cov}[x(k); x(k)/Z(k)].
\]  

(A.28)

It can easily be shown that

\[
\Sigma(k/k) = \sum_{i=1}^{M} P_i(k) [\Sigma_i(k/k) + (\hat{x}_i(k/k) - \hat{x}(k/k)) 
\]

(A.29)

\[
\cdot (\hat{x}_i(k/k) - \hat{x}(k/k))^T].
\]

Note however, that the covariance \( \Sigma(k/k) \) can no longer be calculated off-line, as for a standard Kalman filter, because it is a function of the real time estimates \( \hat{x}_i(k/k) \).
and the a posteriori probabilities $P_i(k)$.

Parameter Identification

$$\theta(k) = E(\theta(k)/Z(k))$$

$$= \int \theta p(\theta/Z(k))d\theta$$

where $p(\theta/Z(k)) = \sum_{i=1}^{M} P_i(k)\delta(\theta-\theta_i)$. (A.31)

Therefore, the parameter estimate can be expressed as:

$$\hat{\theta}(k) = \sum_{i=1}^{M} \theta_i P_i(k).$$ (A.32)

the Bayesian weighted average (by the a posteriori probabilities $P_i(k)$) of all the possible values of the parameter.

It is important at this point to discuss the asymptotic properties of Equations (A.27) and (A.32). If the system is subject to a persistent excitation, then the residuals of the matched Kalman filter will be "small," while the residuals of all the other mismatched filters will be large. Therefore, the a posteriori probability $P_i(k)$ associated with the matched (correct) model will increase, while the mismatched model a posteriori probabilities $P_j(k)$ where $(j \neq i$, etc.)
j=1, 2, ..., M) will decrease. Therefore for large k, (A.27) and (A.32) reduce to

\[ k(k/k) \]

(A.33)

\[ \hat{\theta}_i(k) = \theta_1 \]

(A.34)

where the \( i \)th model is the matched (correct) model.

A.2 Brief Review of the Viterbi Algorithm

In this subsection of Appendix A a brief review of the fundamental properties and structure of the optimal Viterbi algorithm is presented.

Viterbi obtained tight upper and lower bounds for true codes over memoryless channels and formulated a decoding algorithm that was asymptotically optimum. Forney [6] showed that the Viterbi algorithm is actually a maximum likelihood decoder, which is always optimum.

However, the standard approach utilized by Viterbi will not be used. Instead, since this thesis report is primarily concerned with the state-space description and realization of dynamical systems, Omura's [36] state-space formulation of the algorithm is used. Omura [36] suggests that the Viterbi algorithm can be considered a forward dynamic programming solution to a generalized regulator control problem. By viewing a true encoder as a discrete-time dynamical system driven by an information source, he
derives the backward version of the Viterbi algorithm.

Encoder

Consider an information source that generates an output \( u(k-1) \) at time \( k \). The source feeds into an encoder whose output is \( y(k) \). The state of the encoder at time \( k \) is the vector \( s(k) \) which consists of \( j \) elements \([u(k-1), u(k-2), \ldots, u(k-j)]^T\) where \( j \) is the encoder constraint length.

Therefore, the encoder can be represented by the following pair of state equations:

**State Dynamics**

\[
_s(k+1) = E_s(k) + F u(k).
\]  
(A.35)

where \( s(k)= \begin{bmatrix} u(k-j) \\ \vdots \\ u(k-2) \\ u(k-1) \end{bmatrix} ; s(k) \) is a \((j\times1)\) vector  

\[
E = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix} ; E \) is a \((j\times j)\) matrix  

(A.37)
F = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}; F \text{ is a } (j \times 1) \text{ matrix} \quad (A.38)

**Decoder**

The channel output at time $k$ is denoted by $r(k)$.

Thus, for an input sequence $y = [y(1), y(2), \ldots, y(k), \ldots, y(N)]$, the output sequence of the channel is \( r = [r(1), r(2), \ldots, r(k), \ldots, r(N)] \). Let $J(y)$ be defined as the negative log likelihood of the channel output sequence $r$, i.e.,

$$J(y) = -\ln P[r/y] = \sum_{k=1}^{N} \ln P[r(k)/y(k)] \quad (A.39)$$

If all the information sequences are equally likely, the minimum-error-probability decoder chooses the following input sequence estimate $\hat{u}(1), \hat{u}(2), \ldots, \hat{u}(k-1), \ldots$ that minimizes $J(y)$. 
APPENDIX B

DERIVATION OF THE ADAPTIVE JOINT DETECTION
ESTIMATION ALGORITHM (AJDEA)

The purpose of this appendix is to present in detail
the derivation of the Adaptive Joint Detection Estimation
Algorithm introduced in Chapter 3. The dynamical system
that will be used in the formulation is the "limited"
general case that encompasses the three specific cases
outlined in Section 2.2. The "limited" general case was
defined in Section 2.1 by the following pair of state
equations:

\[
\begin{align*}
x(k+1) &= x(k) + w(k) \\
z(k) &= C(k)x(k) + v(k)
\end{align*}
\]  

(B.1)  

(B.2)

where the observation matrix \(C(k)\) is dependent on an unknown
time-varying parameter \(u(k)\) that is either 1 at each time
increment with equal a priori probability.

In Chapter 3 it was emphasized that the exponential
growth of the bank of Kalman filters (Figure 3.1) must be
delimited. The crux of the delimiting operation was shown
to be the use of a weighted average of the Kalman filter
outputs instead of the elemental filter outputs (see Figure
3.2)
Figure B.1 is the generalization of Figure 3.2 to time increments \((k-1)\) and \((k)\). For time increment \(k\) the output of the Kalman filter matched to \(H_1\) (denoted by \(KF/H_1\)) is \(\hat{x}_1(k/k)\), where \(H_1\) is the hypothesis that \(u(k) = 1\). Using standard Kalman filter equations and Figure B.1 it can be shown that

\[
\hat{x}_1(k/k) = \hat{x}(k-1/k-1) + K_1(k)[z(k) - C_1(k)\hat{x}(k-1/k-1)]\quad (B.3)
\]

Similarly the output of the Kalman filter matched to \(H_0\) (denoted by \(KF/H_0\)) is \(\hat{x}_0(k/k)\), where \(H_0\) is the hypothesis that \(u(k) = -1\). It is given by

\[
\hat{x}_0(k/k) = \hat{x}(k-1/k-1) + K_0(k)[z(k) - C_0(k)\hat{x}(k-1/k-1)]\quad (B.4)
\]

The matrices \(C_1(k)\) and \(C_0(k)\) correspond respectively to \(H_1\) and \(H_0\), i.e., under hypothesis \(H_1\) the elements of \(C(k)\) equal to \(u(k)\) are set to 1, while under hypothesis \(H_0\) they are set to -1. All other elements of the matrix \(C(k)\) are either known a priori or are preceding bits that have been estimated.

The next step is to obtain the overall state estimate \(\hat{x}(k/k)\) by summing the weighted filter outputs \(P_1(k)\hat{x}_1(k/k)\) and \(P_0(k)\hat{x}_0(k/k)\).

Therefore,

\[
\hat{x}(k/k) = P_1(k)\hat{x}_1(k/k) + P_0(k)\hat{x}_0(k/k)\quad (B.5)
\]

Substitution of Equations (B.3) and (B.4) into (B.5)
Figure B.1  DELIMITED BANK OF KALMAN FILTERS
results in the following equation for \( \hat{x}(k/k) \):

\[
\hat{x}(k/k) = \hat{x}(k-1/k-1) + P_1(k)K_1(k)[z(k) - C_1(k)\hat{x}(k-1/k-1)]
+ P_0(k)K_0(k)[z(k) - C_0(k)\hat{x}(k-1/k-1)] \tag{B.6}
\]

The term \([z(k) - C_1(k)\hat{x}(k-1/k-1)]\) is the residual of KF/H_1, while \([z(k) - C_0(k)\hat{x}(k-1/k-1)]\) is the residual of KF/H_0. Define

\[
\mathbf{\nabla}_1(k) = \Delta \equiv \begin{bmatrix} z(k) - C_1(k)\hat{x}(k-1/k-1) \\ \vdots \\ z(k) - C_0(k)\hat{x}(k-1/k-1) \end{bmatrix} \tag{B.7}
\]

Equations (B.3) and (B.4) can be rewritten as follows:

\[
\hat{x}_1(k/k) = \hat{x}(k-1/k-1) + K_1(k)\mathbf{\nabla}_1(k) \tag{B.9}
\]

\[
x_0(k/k) = x(k-1/k-1) + K_0(k)\mathbf{\nabla}_0(k) \tag{B.10}
\]

and Equation (B.6) as follows:

\[
x(k/k) = x(k-1/k-1) + P_1(k)K_1(k)\mathbf{\nabla}_1(k) + P_0(k)K_0(k)\mathbf{\nabla}_0(k) \tag{B.11}
\]

Equation (B.11) is a recursive update equation of the delimited state estimate. However, it poses the computational burden of implementing two separate Kalman filters. This burden can be appreciable for complex systems like the Multipath Communication Case. Hence, we choose to further
reduce the required computation by proposing the following two distinct alternatives.

Alternative 1 consists mainly of a standard maximum likelihood detection test. Only the hypothesis with the larger a posteriori probability \( P_1(k) \) is retained. Therefore, only one Kalman filter actually has to be implemented.

Alternative 2 is based on a distinctly different approach. Namely, the alternative entails using the weighted residual \( r(k) \), i.e.,

\[
  r(k) \triangleq P_1(k)r_1(k) + P_0(k)r_0(k) \tag{B.12}
\]

instead of the elemental residuals \( r_1(k) \) and \( r_2(k) \) to drive the pair of conditioned Kalman filters. In particular this scheme implies that the elemental state estimate update Equations (B.9) and (B.10) are modified as follows:

\[
  \hat{x}_1(k/k) = \hat{x}(k-1/k-1) + K_1(k)r(k) \tag{B.13}
\]

\[
  \hat{x}_0(k/k) = \hat{x}(k-1/k-1) + K_0(k)r(k) \tag{B.14}
\]

This is a crucial step which deserves careful attention. A
viable alternative to parallel processing of the information contained in the elemental residuals \( r_1(k) \) and \( r_2(k) \) is to filter their appropriately weighted sum.

Substituting for \( r_1(k) \) and \( r_0(k) \) in Equation (B.12) and recalling that \( P_1(k) + P_0(k) = 1 \), we obtain

\[
r(k) = z(k) - [P_1(k)C_1(k) + P_0(k)C_0(k)]\hat{x}(k-1/k-1) \quad \text{(B.15)}
\]

Note that the coefficient of \( \hat{x}(k-1/k-1) \) is the weighted sum of the two possible values of \( \hat{C}(k) \). Let \( \hat{C}(k) \) denote this weighted sum, i.e.,

\[
\hat{C}(k) = P_1(k)C_1(k) + P_0(k)C_0(k) \quad \text{(B.16)}
\]

Therefore Equation (B.15) simplifies to

\[
r(k) = z(k) - \hat{C}(k)\hat{x}(k-1/k-1) \quad \text{(B.17)}
\]

which is intuitively reassuring. The weighted residual is essentially the difference between the measurement \( z(k) \) and the best predicted measurement \( \hat{C}(k)\hat{x}(k-1/k-1) \) that can be made a priori. Based on Equation (B.17) the two elemental update equations can be rewritten as

\[
\hat{x}_1(k/k) = \hat{x}(k-1/k-1) + K_1(k)[z(k) - \hat{C}(k)\hat{x}(k-1/k-1)] \quad \text{(B.18)}
\]

\[
\hat{x}_0(k/k) = \hat{x}(k-1/k-1) + K_0(k)[z(k) - \hat{C}(k)\hat{x}(k-1/k-1)] \quad \text{(B.19)}
\]

At this point let us pause and reflect on what has been
accomplished thus far. A pair of equations has been derived that generates the update state estimates of the two conditioned Kalman filters. The estimates are based on the measurement $z(k)$, the preceding weighted state estimate $\hat{x}(k-1/k-1)$, and the time dependent weighting coefficients $P_1(k)$ and $P_0(k)$. (Note the only difference between Equations (B.18) and (B.19) are the gains $K_1(k)$ and $K_0(k)$).

Substituting Equations (B.13) and (B.14) into Equation (B.5) results in the following equation for $\hat{x}(k/k)$:

$$
\hat{x}(k/k) = \hat{x}(k-1/k-1) + [P_1(k)K_1(k) + P_0(k)K_0(k)] \\
\cdot [z(k) - C(k)x(k-1/k-1)]
$$

(B.20)

For notational brevity let

$$
\tilde{G}(k) = [P_1(k)K_1(k) + P_0(k)K_0(k)] 
$$

(B.21)

i.e., $\tilde{G}(k)$ is the weighted average of the gains of the two Kalman filters. Thus (B.20) can be written as

$$
\hat{x}(k/k) = [I - \tilde{G}(k)\tilde{C}(k)]\hat{x}(k-1/k-1) + \tilde{G}(k)z(k)
$$

(B.22)

This equation is almost identical to the standard Kalman filter state estimate update Equation (A.9). The only difference is that for this system the observation matrix is unknown and thus a weighted estimate $\hat{C}(k)$ is used. Thus far, we have successfully delimited the sequential
growth of the bank of filters to two filters and in turn by feeding them the weighted residual vector, the bank of two filters has been compressed to a single aggregate filter.

The structure of the estimation process of the AJDEA is presented in Figure B.2. Note particularly the use of the estimate \( \hat{C}(k) \) and that the predicted measurement is given by

\[
\hat{x}(k-1/k-1) = \hat{C}(k)\hat{x}(k-1/k-1) .
\]  

The next step in the derivation of the AJDEA is the formulation of update equations for the gain matrix \( G(k) \) and the error covariance \( E(k/k) \).

**Error Covariance \( E(k/k) \)**

Let \( \tilde{G}(k) \) represent the difference between the actual observation matrix and the estimate of the observation matrix, i.e.,

\[
\tilde{G}(k) = G(k) - \hat{G}(k) \tag{B.24}
\]

where \( \tilde{G}(k) \) may have been generated by either Alternative 1 or 2. Similarly, let \( \tilde{e}(k) \) be the incremental error made in state estimation, i.e.,

\[
\tilde{e}(k) = \tilde{x}(k/k) - x(k) \tag{B.25}
\]
Figure B.2 ESTIMATION STRUCTURE OF AJDEA
Therefore,

\[ e(k) = [I - G(k)\hat{C}(k)]x(k-l/k-1) + G(k)z(k) - x(k-l) - w(k-1) \]  

(B.26)

Moreover, it can be shown by substituting Equation (B.24) into (B.26) that

\[ e(k) = [I - G(k)\hat{C}(k)]e(k-1) + G(k)\hat{C}(k)w(k-1) + v(k) \]

- \( w(k-1) - G(k)\hat{C}(k)x(k-l) \)  

(B.27)

The error covariance matrix \( \Sigma(k/k) \) is given by

\[ \Sigma(k/k) = E[e(k)e^T(k)] \]  

(B.28)

while

\[ \Sigma(k/k-1) = \Sigma(k-1/k-1) + Q(k-1) \]  

(B.29)

where \( Q(k-1) = E[w(k-1)w^T(k-1)] \)

Now let

\[ S(k) = E[x(k)x^T(k)]. \]  

(B.30)

After a little algebraic manipulation it can be shown that

\[ \Sigma(k/k) = [I - G(k)\hat{C}(k)](\Sigma(k-1/k-1) + Q(k-1))[I - G(k)\hat{C}(k)]^T + G(k)[E(\hat{C}(k)S(k-1)\hat{C}^T(k)) + R(k)] \]

+ \( E(\hat{C}(k)Q(k-1)\hat{C}^T(k))][G^T(k) \]

(B.31)
However, for all three cases of interest in this
report, the process noise $\bar{w}(k)$ is "small" with respect
to $\bar{x}(k)$ and its covariance is much "smaller" than the
measurement noise covariance $\bar{R}(k)$, i.e.,

$$w(k) = 0$$

$$||\bar{Q}(k)|| < ||\bar{R}(k)||$$

Thus the state covariance matrix can be approximated
by

$$\Sigma(k) = E[\bar{x}(k)\bar{x}^T(k)] = \bar{x}(k)\bar{x}^T(k)$$

(B.33)

Next by substituting (B.33) into (B.31) a simplified
expression for $\zeta(k/k)$ is obtained

$$\zeta(k/k) = (I - G(k)\hat{\bar{C}}(k))(\zeta(k-1/k-1) + \bar{Q}(k-1))[I - G(k)\hat{\bar{C}}(k)]^T$$

$$+ G(k)[E(\bar{C}(k)\bar{x}(k-1)\bar{x}^T(k-1)\bar{C}^T(k)) + \bar{R}(k)]G^T(k)$$

(B.34)

**Gain Matrix $G(k)$**

The next step is the derivation of a recursive equation
for $G(k)$, the filter gain matrix. The criterion for deter-
mining $G(k)$ is the minimization of a weighted scalar sum of
the diagonal elements of the state error covariance matrix
$\Sigma(k/k)$. Therefore, choose $J(k)$ to be the cost function
defined by
\[ J(k) = E[e^T(k)M_0e(k)] \]  
\[ \text{(B.35)} \]

where \( M \) is an arbitrary semi-definite matrix. It is shown by Gelb [7] that the optimal estimate is independent of \( M \). Thus, let \( M=I \).

Hence, the cost function is reduced to

\[ J(k) = E[e^T(k)e(k)] = \text{trace}[\Sigma(k/k)]. \]  
\[ \text{(B.36)} \]

The next step is to obtain \( G(k) \) by minimizing \( J(k) \). This is done by taking the partial derivative of \( J(k) \) with respect to \( G(k) \) and setting it equal to zero, i.e.,

\[ \frac{\partial}{\partial G(k)} [\text{trace} \Sigma(k/k)] = -2[I-G(k)\hat{g}(k)](\hat{z}(k-1/k-1)+Q(k-1)\hat{g}^T(k)) \]
\[ = \frac{\partial G(k)}{\partial G(k)} [E(\hat{g}(k)\hat{z}(k-1)\hat{g}^T(k)) + R(k)] \]
\[ + E(\hat{g}(k)Q(k-1)\hat{g}^T(k))] = 0. \]  
\[ \text{(B.37)} \]

Solving for \( G(k) \) gives the following equation:

\[ G(k) = (\Sigma(k-1/k-1)+Q(k-1))\hat{g}^T(k)\hat{z}(k)(E(\hat{g}(k)\hat{z}(k-1)+Q(k-1))\hat{g}^T(k)) \]
\[ + E(\hat{g}(k)\hat{z}(k-1)\hat{g}^T(k))+R(k)+E(\hat{g}(k)Q(k-1)\hat{g}^T(k))]^{-1}. \]  
\[ \text{(B.38)} \]

Once again, as in the formulation of \( \Sigma(k/k) \), the update equation greatly simplifies for the case, where \( \Sigma(k) \leq \Xi \) and \( \|z(k)\| < R(k) \), to the following equation:

\[ G(k) = (\Sigma(k-1/k-1)+Q(k-1))\hat{g}^T(k)\hat{z}(k)(E(\hat{g}(k)\hat{z}(k-1)+Q(k-1))\hat{g}^T(k)) \]
\[ + E(\hat{g}(k)\hat{z}(k-1)\hat{g}^T(k))+R(k)+E(\hat{g}(k)Q(k-1)\hat{g}^T(k))]^{-1}. \]  
\[ \text{(B.39)} \]
The derivation of the predict cycle and update cycle of the estimation part of the AJDEA is complete. For easy future reference a list of the equations is given below.

**Predict Cycle**

\[
\hat{x}(k/k-1) = \hat{x}(k-1/k-1) \tag{B.40}
\]

\[
\Sigma(k/k-1) = \Sigma(k-1/k-1) + Q(k-1) \tag{B.41}
\]

**Update Cycle**

\[
\hat{x}(k/k) = [I - G(k)C(k)]\hat{x}(k-1/k-1) + \Sigma(k)x(k) \tag{B.42}
\]

\[
\Sigma(k/k) = [I - G(k)C(k)]\Sigma(k-1/k-1)[I - G(k)C(k)]^T
+ G(k)[E(C(k)x(k-1)x^T(k-1)C^T(k))] + R(k) \tag{B.43}
\]

\[
G(k) = \Sigma(k/k-1)C^T(k)[C(k)\Sigma(k-1/k-1)C^T(k)]^T
+ E(C(k)x(k)C^T(k)) + R(k)]^{-1} \tag{B.44}
\]

Equations (B.40) - (B.44) are similar to the standard Kalman filter equations except for the use of the estimate \(\hat{C}(k)\) instead of the unknown \(C(k)\) and the extra state dependent term \(E(C(k)x(k)x^T(k)C^T(k))\). The latter is a major computational penalty. Obviously this term cannot be evaluated \textit{a priori}. Fortunately for most cases of interest in this
thesis it does not have to be evaluated explicitly. It can be accounted for by suitably inflating the measurement noise covariance. An intuitive argument supporting this statement is presented below. For further discussion of this state dependent term the reader is referred to Gustafson and Speyer [9].

In Chapter 3 it was pointed out that in this thesis we are interested in two classes of signal to noise ratio conditions: "high to moderate" and "low." At first glance one may postulate that the contribution of \( E[C(k)xx^T C^T(k)] \) is negligible for high SNR since detection errors are infrequent and that the significance of this term becomes more pronounced for lower SNR conditions. However, it shall be shown by using an intuitive argument that the contribution is significant and comparable for both classes. The intuitive argument is based on the analysis of a hypothetical example.

The example is taken from the Two-Dimensional Vector Case that was defined in Chapter 2. Namely, the stochastic system is given by

\[
x(k+1) = x(k) \tag{B.45}
\]

\[
z(k) = C(k)x(k) + v(k) \tag{B.46}
\]

where - the state vector \( x(k) = [x_1(k) x_2(k)]^T \).
- the unknown observation matrix
\[ \mathbf{C}(k) = [u(k) \quad u(k-1)]^T \] (B.48)

- the measurement \( z(k) \) is a scalar

- data bit \( u(k) = \begin{cases} +1 & P(u(k)=1)=p \\ -1 & P(u(k)=-1)=q \end{cases} \) (B.49)

Consider applying Alternative 1 of the AJDEA to this system.

Let \( \mathbf{C}(k) \) denote the error in the observation matrix estimate, i.e.,

\[ \mathbf{C}(k) = [\tilde{u}(k) \quad \tilde{u}(k-1)] \] (B.50)

where \( \tilde{u}(k) \) the data bit estimate error, i.e.,

\[ \tilde{u}(k) = u(k) - \hat{u}(k) \] (B.51)

Clearly \( \tilde{u}(k) \) is a random variable with the probability mass function presented in Figure B.3 where \( p \) is the probability of correct detection, while \( q \) is the probability of incorrect detection, and \( p + q = 1 \). Moreover, it can be shown that the expected value and second moment of \( \tilde{u}(k) \) are

\[ E(\tilde{u}(k)) = 0 \] (B.52)

\[ E(\tilde{u}^2(k)) = 4q \] (B.53)
Figure B.3 PROBABILITY MASS FUNCTION OF THE DATA BIT ESTIMATION ERROR \( \tilde{u}(k) \) GENERATED BY ALTERNATIVE 1

\[
E[\tilde{u}(k)] = 0 \\
E[|\tilde{u}(k)|^2] = 4q
\]
The next step is the evaluation of the term $E[\tilde{C}(k)xx^T\tilde{C}(k)]$ by substituting the values from Equation (B.47) and (B.50). Hence

$$E[\tilde{C}(k)xx^T\tilde{C}(k)] \text{ reduces to }$$

$$E\left[(\tilde{u}(k) \quad \tilde{u}(k-1))\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \begin{pmatrix} x_1 & x_2 \\ \tilde{u}(k) & \tilde{u}(k-1) \end{pmatrix}\right]$$

$$= E[\tilde{u}^2(k)x_1^2 + 2\tilde{u}(k)\tilde{u}(k-1)x_1x_2 + \tilde{u}^2(k-1)x_2^2]. \quad (B.54)$$

Assuming that $\tilde{u}(k)$ and $\tilde{u}(k-1)$ are uncorrelated and using Equations (B.52) and (B.53) we obtain

$$E[\tilde{C}(k)xx^T\tilde{C}(k)] = E[\tilde{u}^2(k)x_1^2 + \tilde{u}^2(k-1)x_2^2]$$

$$= 4q(x_1^2 + x_2^2). \quad (B.55)$$

Now consider the following two hypothetical cases. The first falls into the "high to moderate" SNR class, while the second falls into the "low" SNR class.

Case 1: Number of detection errors in identifying a data bit sequence of fifty bits is 1.

Based on this information the best estimate of $q$ (the probability of a detection error for a given bit) one can make is (.02).
Hence, from Equation (B.55) the average contribution of \( E[\hat{C}(k)xx^{T}\hat{C}^{T}(k)] \) is \( .08(x_1^2 + x_2^2) \).

**Case 2:** Number of detection errors in identifying a data bit sequence of fifty bits is 10.

Based on this information the best estimate of \( q \) one can make is \(.2\). Hence from Equation (B.55) the average contribution of \( E[\hat{C}(k)xx^{T}\hat{C}^{T}(k)] \) is \(.8(x_1^2 + x_2^2)\).

The next step is to determine the implications of these results on the update covariance matrix \( \Sigma(k/k) \).

From Equation (B.34) we know that

\[
\Sigma(k/k) = [I - G(k)\hat{C}(k)]\Sigma(k/k-1)[I - G(k)\hat{C}(k)]^{T} \\
+ G(k)[E(\hat{C}(k)xx^{T}\hat{C}^{T}(k)) + R]G^{T}(k) \tag{B.56}
\]

Let

\[
R_T(k) = E[\hat{C}(k)xx^{T}\hat{C}^{T}(k)] + R \tag{B.57}
\]

Hence, the effect of \( E[\hat{C}(k)xx^{T}\hat{C}^{T}(k)] \) on the error covariance \( \Sigma(k/k) \) for this Two-Dimensional Vector Case is to increase the measurement noise covariance matrix. From Case 1 we know that the increase for a hypothetical "high to moderate" SNR case is \(.08(x_1^2 + x_2^2)\), while from Case 2 we obtained
.8(x_1^2 + x_2^2) for a "low" SNR case. A likely value of the state vector of Case 1 is \( x = [2.0 \ 2.0]^T \), and assume the measurement covariance \( R = .5 \). Therefore, the increase is 

\[
.08(8) = (.64). \]

Hence, \( R_T \) which was defined in Equation (B.57) equals (1.14).

Now let us consider a likely value for the state vector of Case 2. Assume \( x = [.70 \ .70]^T \) and once again \( R = .5 \). The increase from \( E[\bar{C}(k)xx^T\bar{C}^T(k)] \) is .784. Hence \( R_T \) equals (1.284).

Note that the contribution of \( E[\bar{C}(k)xx^T\bar{C}^T(k)] \) under both "high to moderate" and "low" SNR conditions are comparable and must be accounted for. The technique that was used in the computer simulations performed for this thesis was to "inflate" the noise covariance \( R \). "Engineering" judgment must be used in determining how much to increase the value of \( R \). Computer simulation results presented in Chapters 4, 5, and 6 substantiate this technique.

The conclusion therefore, is to increase the measurement noise covariance. Intuitively this is appealing since the general class of systems addressed in this thesis are situated on a higher level of complexity than the systems encountered in standard linear filtering theory. The overall measurement error covariance should reflect the uncertainty associated with the unknown identity of the transmitted data bits. Thus, hereafter we shall consider \( R_T \) defined in Equation (B.57) to
be the "overall" measurement error covariance.

In order to avoid complicating the notation unnecessarily the subscript "T" shall be omitted hereafter, although understood in appropriate places. Therefore, the two Equations (B.43) and (B.44) can be expressed as

\[
\Sigma(k/k) = [I - G(k)\hat{\theta}(k)]\Sigma(k/k-1)[I - G(k)\hat{\theta}(k)]^T \\
+ G(k)R(k)G^T(k) \\
G(k) = E(k/k-1)C(k)\Sigma(k/k-1)C^T(k) + R(k) \\
\text{ (B.58)}
\]

A more simplified state error covariance matrix update equation can be obtained by substituting Equation (B.59) into (B.58) and performing a few simple algebraic manipulations. Thus

\[
\Sigma(k/k) = \Sigma(k/k-1) - G(k)\hat{\theta}(k)\Sigma(k/k-1) - \Sigma(k/k-1)\hat{\theta}^T(k)G^T(k) \\
+ G(k)[\hat{\theta}(k)\Sigma(k/k-1)\hat{\theta}^T(k) + R(k)]G^T(k) \\
\text{which simplifies to}
\]

\[
\Sigma(k/k) = [I - G(k)\hat{\theta}(k)]\Sigma(k/k-1) \\
\text{ (B.60)}
\]

Note that unlike the standard Kalman filter problem \(\Sigma(k/k)\) cannot be precomputed off-line since it contains the real time state estimate \(\hat{x}(k/k)\) and the observation matrix estimate \(\hat{\theta}(k)\) which is a function of the real time measure-
ments.

The derivation of the Adaptive Joint Detection Estimation Algorithm is complete. Listed below are estimator equations:

**Predict Cycle**

\[
\hat{x}(k/k-1) = \hat{x}(k-1/k-1) \quad (B.61)
\]

\[
\Sigma(k/k-1) = \Sigma(k-1/k-1) + Q(k-1) \quad (B.62)
\]

**Update Cycle**

\[
\hat{x}(k/k) = \hat{x}(k/k-1) + G(k)[z(k) - \hat{\Theta}(k)\hat{x}(k/k-1)] \quad (B.63)
\]

\[
\Sigma(k/k) = [I - G(k)\hat{\Theta}(k)]\Sigma(k/k-1) \quad (B.64)
\]

\[
G(k) = \Sigma(k/k-1)\hat{\Theta}^T(k)[\hat{\Theta}(k)\Sigma(k/k-1)\hat{\Theta}^T(k) + R(k)]^{-1} \quad (B.65)
\]
APPENDIX C

ANALYTIC PERFORMANCE BOUND OF AJDEA-ALTERNATIVE 1 FOR THE SCALAR CASE

Recall from Chapter 4 Equation (4.18) that the state estimate assuming unity initial covariance is given by

\[ \hat{x}(k/k) = \frac{\sum_{i=0}^{k} \hat{u}(i)z(i)}{1 + k}. \quad (C.1) \]

In this appendix it is shown that for reasonable SNR conditions

\[ \lim_{k \to \infty} E[\hat{x}(k/k)] = |x|. \quad (C.2) \]

The expected value of \( \hat{x}(k/k) \) is given by

\[ E(\hat{x}(k/k)) = \frac{E\left\{ \sum_{i=1}^{k} \hat{u}(i)z(i) \right\}}{1 + k}. \quad (C.3) \]

\[ = \frac{\sum_{i=1}^{k} E(|z(i)|)}{1 + k}. \]

-205-
Now we need to calculate $E(|z(i)|)$. Let $p_z(z_0)$ be the density of the measurement $z$. It can be expressed as follows

$$p_z(z_0) = \int p(z, H) dH$$

\[ \text{(C.4)} \]

$$= \int p(z/H)p(H) dH$$

where $H \in \{H_0, H_1\}$. $H_0$ is the hypothesis that $u(k)=-1$, while $H_1$ is the hypothesis that $u(k)=1$, i.e.,

$$P(H) = P(H=H_0)\delta(H-H_0) + P(H=H_1)\delta(H-H_1)$$

where $\delta(\cdot)$ is the Kroenecker delta function.

Now assuming $P(H=H_0) = P(H=H_1) = \frac{1}{2}$, Equation (C.4) can be rewritten

$$p(z_0) = \int_{-\infty}^{\infty} p(z/H) \left[ \frac{\delta(H-H_0)}{2} + \frac{\delta(H-H_1)}{2} \right] dH. \text{ (C.5)}$$

Therefore

$$E(|z(i)|) = \int_{-\infty}^{\infty} |z(i)| \left[ \frac{1}{2}p(z/H_0) + \frac{1}{2}p(z/H_1) \right] dz. \text{ (C.6)}$$

Since $z(k) = u(k)x(k) + v(k)$ and $v(k)$ is Gaussian, then

$$E(|z(i)|) = \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} \left[ e^{-(z+x)^2/2\sigma^2} + e^{-(z-x)^2/2\sigma^2} \right] dz \text{ (C.7)}$$
where \( r \) is the variance of the noise \( v(k) \).

Equation (C.7) can be expanded to

\[
E\{ z(1) \} = \frac{1}{2} \int_{-\infty}^{0} \frac{(-z)}{\sqrt{2\pi r}} e^{-(z+x)^2/2r} \, dz + \frac{1}{2} \int_{0}^{\infty} \frac{(z)}{\sqrt{2\pi r}} e^{-(z+x)^2/2r} \, dz + \frac{1}{2} \int_{0}^{\infty} \frac{(-z)}{\sqrt{2\pi r}} e^{-(z-x)^2/2r} \, dz + \frac{1}{2} \int_{-\infty}^{0} \frac{(z)}{\sqrt{2\pi r}} e^{-(z-x)^2/2r} \, dz
\]

(Labeled as (C.8)

Let \( a = \frac{1}{2} \int_{0}^{\infty} (-2z)e^{-(z+x)^2/2r} \, dz + \frac{1}{2} \int_{-\infty}^{0} (2z)e^{-(z-x)^2/2r} \, dz \)

and add to both sides of Equation (C.8). After some manipulation it can be shown that

\[
E\{ |z(1)| \} = |x| - a.
\]

(Labeled as (C.9)

Now we must evaluate \( a \). Once again, after some algebraic manipulation, it can be shown:

\[
a = \frac{2}{\sqrt{2\pi r}} \int_{-\infty}^{0} z e^{-(z-x)^2/2r} \, dz.
\]

(Labeled as (C.10)

Now letting \( y = z-x \) and performing the variable transformation \( m = \frac{y}{\sqrt{r}} \), we obtain:
Finally, after numerous other variable transformations, we obtain

\[
a = 2 \int_{-\infty}^{\frac{-x}{\sqrt{r}}} m \frac{\sqrt{r}}{2\pi} e^{-m^2/2} \, dm + x \int_{-\infty}^{\frac{x}{\sqrt{r}}} e^{-m^2/2} \, dm
\]

where \( \phi\left(\frac{x}{\sqrt{r}}\right) = \int_{-\infty}^{\frac{x}{\sqrt{r}}} \frac{1}{\sqrt{2\pi}} e^{-m^2/2} \, dm. \)

As expected, as \( x \) gets larger for fixed statistics of measurement noise (i.e., the SNR increases), the scalar \( a \) goes to zero.

Hence, from Equation (C.9) we know that

\[
E(|z(k)|) = |x|
\]

Therefore

\[
E(\hat{x}(k/k)) = \frac{\sum_{i=1}^{k} |x|}{1 + k}
\]  

(C.13)
Clearly, as $k$-large

$$E(\hat{x}(k/k)) = |x|.$$  \hspace{1cm} (C.14)
APPENDIX D

DERIVATION OF THE ERROR COVARIANCE MODIFICATION

In Section 7.1, a modification to the error covariance matrix of Alternative 2 was formulated. In this appendix the detailed steps are provided for proceeding from Equation (7.15) to (7.16).

Hence, from Equation (7.15) we know that

\[
\Sigma(k/k) = \sum_{i=0}^{1} P_i(k)E\{[\hat{x}(k)-\hat{x}(k/k)][\hat{x}(k)-\hat{x}(k/k)]^T\} \quad (D.1)
\]

\[
= \sum_{i=0}^{1} P_i(k)E\{x(k)x^T(k)\} - \sum_{i=0}^{1} P_i(k)E\{\hat{x}(k/k)\hat{x}(k/k)\} + \sum_{i=0}^{1} P_i(k)E\{\hat{x}(k/k)\hat{x}(k/k)\}.
\]

In order to perform the necessary algebraic manipulations in an orderly fashion, the four terms on the right hand side of Equation (A.2) are enumerated 1, 2, 3, 4.

Reduction of 1

From Equations (7.7) and (7.8) we know that
\[ x(k) = \tilde{x}_k(k/k) + \tilde{x}_k(k/k). \]  

(D.3)

It can easily be demonstrated, therefore, that term \( Q \) reduces to the following expression by substituting for \( x(k) \) in Equation (D.2). Hence,

\[
\sum_{i=0}^{1} P_i(k)\left[ \tilde{x}_i(k/k) + \hat{x}_i(k/k) \right] [\tilde{x}_i(k/k) + \hat{x}_i(k/k)]^T
\]

\[
= \sum_{i=0}^{1} P_i(k) E\{\tilde{x}_i(k/k)\tilde{x}_i^T(k/k)\} + \sum_{i=0}^{1} P_i(k)\hat{x}_i(k/k)\hat{x}_i^T(k/k)
\]

\[
= \sum_{i=0}^{1} P_i(k) \Sigma_i(k/k) + \sum_{i=0}^{1} P_i(k)\hat{x}_i(k/k)\hat{x}_i^T(k/k). \quad (D.4)
\]

Next we need to determine an "update" equation for \( \Sigma_1(k/k) \) in terms of \( \Sigma(k/k-1) \). Essentially, by using the same steps as in Appendix B, it can be shown that

\[
\Sigma_1(k/k) = E\{\tilde{x}_1(k)\tilde{x}_1^T(k)\}
\]

\[
= [I-K_1(k)\hat{\theta}(k)]\Sigma(k/k-1)[I-K_1\hat{\theta}(k)]
\]

\[
+ K_1(k)R(k)K_1^T(k)
\]

\[
= [I-K_1(k)\hat{\theta}(k)]\Sigma(k/k-1). \quad (D.5)
\]
where \( R(k) \) is the "inflated" measurement noise covariance matrix that is discussed in Appendix B.

Substituting for \( \hat{z}_i(k) \) in Equation (D.4) with Equation (D.5) and remembering that the overall gain \( G(k) \) was defined earlier as the weighted sum of \( K_1(k) \) and \( K_0(k) \), i.e.,

\[
G(k) = P_1(k)K_1(k) + P_0(k)K_0(k)
\]  \hspace{1cm} (D.6)

we obtain the following expression for term \( \Theta \).

\[
\left[ I - G(k)\overset{\hat{\Sigma}}{\Sigma}(k/k) \right] \sum(k/k-1) + \sum_{i=0}^{1} P_1(k)\hat{x}_i(k/k)\hat{x}_i^T(k/k).
\]  \hspace{1cm} (D.7)

Reduction of \( \Theta \)

\[
\sum_{i=0}^{1} P_1(k)E\{\hat{x}(k/k)\hat{x}^T(k/k)\} = \sum_{i=0}^{1} P_1(k)E\{\hat{x}(k/k)[\hat{x}_i(k/k)+\hat{x}_i(k/k)]^T\}
\]

\[
= \sum_{i=0}^{1} P_1(k)\hat{x}(k/k)\hat{x}_i^T(k/k)
\]

\[
= \hat{x}(k/k)\sum_{i=0}^{1} P_1(k)\hat{x}_i^T(k/k)
\]

\[
= \hat{x}(k/k)\hat{x}_1^T(k/k).
\]  \hspace{1cm} (D.8)
Reduction of $\Omega$

It can be easily shown that

$$\sum_{i=0}^{1} P_i(k) E\{X(k)\hat{X}_i^T(k/k)\} = \hat{X}(k/k)\hat{X}^T(k/k). \quad (D.9)$$

Reduction of $\Psi$

Similarly

$$\sum_{i=0}^{1} P_i(k) \hat{\Psi}(k/k)\hat{\Psi}_i^T(k/k) = \hat{\Psi}(k/k)\hat{\Psi}^T(k/k) \sum_{i=0}^{1} P_i(k) \quad (D.10)$$

$$= \hat{\Psi}(k/k)\hat{\Psi}^T(k/k).$$

Now, combining all four terms we obtain:

$$\Sigma(k/k) = [I - G(k)\hat{C}(k)]\Sigma(k/k-1) + \left( \sum_{i=0}^{1} P_i(k)\hat{\Xi}_i(k/k)\hat{\Xi}_i^T(k/k) \right)$$

$$- \hat{\Psi}(k/k)\hat{\Psi}^T(k/k). \quad (D.11)$$

Now, recall the three state estimate update equations

$$\hat{\xi}_1(k/k) = \hat{\xi}(k/k-1) + K_1(k)r(k)$$

$$\hat{\xi}_2(k/k) = \hat{\xi}(k/k-1) + K_0(k)r(k) \quad (D.12)$$

$$\hat{\xi}(k/k) = \hat{\xi}(k/k-1) + C(k)r(k).$$

Substituting Equation (D.12) into (D.11) and reducing
yields

\[ E(k/k) = \left[ I - G(k)C(k) \right] E(k/k-1) + \left[ \sum_{i=0}^{1} P_i(k)K_i(k)r(k)r^T(k)K_i^T(k) \right] \]

\[ - G(k)r(k)r^T(k)G^T(k) \]  

(D.13)

where the gains \( K_1(k) \) and \( K_0(k) \) must be updated at each iteration, i.e.,

\[ K_i(k) = \frac{E(k/k-1)C_i^T(k)\left[C_i(k)E(k/k-1)C_i^T(k) + R(k)\right]^{-1}}{} \]  

(D.14)

where \( i \in \{0,1\} \).
REFERENCES


32. Middleton, D. and Esposito, R., "Simultaneous Optimum


