ADAPTIVE PROCESSING IN HIGH
FREQUENCY SURFACE WAVE RADAR

BY

Oliver Saleh

A thesis submitted in conformity with the requirements
for the degree of Master of Applied Science,
Graduate Department of Electrical and Computer Engineering
in the University of Toronto.

Copyright © 2008 by Oliver Saleh.
All Rights Reserved.
Abstract

High-Frequency Surface Wave Radar (HFSWR) is a radar technology that offers numerous advantages for low-cost, medium accuracy surveillance of coastal waters beyond the exclusive economic zone. However, target detection and tracking is primarily limited by ionospheric interference. Ionospheric clutter is characterized by a high degree of nonhomogeneity and nonstationarity, which makes its suppression difficult using conventional processing techniques. Space-time adaptive processing techniques have enjoyed great success in airborne radar, but have not yet been investigated in the context of HFSWR. This thesis is primarily concerned with the evaluation of existing STAP techniques in the HFSWR scenario and the development of a new multistage adaptive processing approach, dubbed the Fast Fully Adaptive (FFA) scheme, which was developed with the particular constraints of the HFSWR interference environment in mind. To this end an analysis of measured ionospheric clutter is conducted, which allows for a characterization of the underlying statistical behavior of the HFSWR interference. Several popular low-complexity STAP algorithms are assessed for their performance in HFSWR. Focus is then shifted towards the development of the FFA approach. Three different spatio-temporal partitioning schemes are introduced. A thorough investigation of the performance of the FFA is conducted in both the airborne radar and HFSWR setups under a variety of sample-support scenarios.
Dedicated to Mariana-Diana and Sami; the best parents a person could hope for.
I would like to thank everyone who contributed to the completion of this work. First and foremost, I’d like to extend my deepest gratitude to Dr. Adve who made this project possible. His guidance, patience, and encouragement laid the foundation of this work; his attention to detail, constructive feedback, and relentless revising was the glue that helped hold everything together. I truly feel privileged to be one of his students.

I would also like to thank Dr. Ravan for all the support and advice she provided throughout the life of this project. Also deserving thanks, is Dr. Riddolls, who provided us with the measured datasets, used in our simulations, and whose helpful feedback helped keep things in context. Thanks also goes to Dr. Plataniotis for his patience, useful comments, and feedback during several meetings with him.

I would also like to thank all my office mates and friends for their support and encouragement throughout my stay in Toronto. It is as Plato once said, “Nature has no love for solitude, and always leans, as it were, on some support; and the sweetest support is found in friendship”.

Last, but certainly not least, I want to extend my humble thanks to my family, who has been there for me since the very beginning. Words would certainly fall short of describing my gratitude, so I will suffice by saying thank you; thank you for everything...

Oliver S. Saleh
June 2008
# Contents

List of Abbreviations ................................................. 2

1 Introduction ...................................................... 1
   1.1 Motivation ................................................ 2
   1.2 System and Data Models .................................. 3
      1.2.1 Airborne Radar System Model ...................... 3
      1.2.2 HFSWR System Model ................................ 6
   1.3 Preliminary Data Analysis: Measured Data ................. 8
      1.3.1 Element-Range Plots ................................ 9
      1.3.2 Range-Doppler Plots ................................ 11
      1.3.3 Angle Doppler Plots ................................ 14
   1.4 Thesis Overview .......................................... 16

2 Processing of HFSWR Returns .................................. 19
   2.1 Review of Fully Adaptive STAP ........................... 20
   2.2 Literature Review of HFSWR Processing Techniques ..... 22
   2.3 Review of Low Complexity STAP Techniques .............. 26
      2.3.1 Joint Domain Localized (JDL) Processing .......... 26
      2.3.2 $D^3$ and Hybrid Approaches ...................... 27
      2.3.3 Parametric Adaptive Matched Filter (PAMF) ....... 29
      2.3.4 Multistage Wiener Filter (MWF) ................... 31
   2.4 Applying STAP to HFSWR ................................... 31
   2.5 Performance of Available STAP Algorithms in HFSWR ..... 33
2.5.1 Realistic Target Model .................................. 33
2.5.2 Simulation Results ...................................... 35

3 Fast Fully Adaptive Processing: Overview and Implementation 42
   3.1 Regular FFA ........................................... 43
   3.2 Interleaved FFA ....................................... 49
   3.3 Unequal Partitions ..................................... 53
   3.4 Randomized FFA ....................................... 56
   3.5 A General Adaptive Multistage Processing Framework ............. 59
   3.6 Complexity Analysis ................................... 60

4 FFA: Numerical Evaluation 68
   4.1 Probability of Detection versus SNR .......................... 69
   4.2 MSMI vs Range ......................................... 77
   4.3 FFA Parameter Selection ................................... 86

5 Conclusions and Future Work 93
   5.1 Conclusion ............................................. 93
   5.2 Future Work ............................................ 94

A Appendix 96
   A.1 Airborne Radar Interference Models .......................... 96
      A.2.1 Joint Domain Localized Processing ..................... 98
      A.2.2 The Direct Data Domain Algorithm .................... 101
      A.2.3 The Hybrid Algorithm ................................ 104
      A.2.4 Parametric Adaptive Matched Filter (PAMF) ........... 105
      A.2.5 Multistage Wiener Filter (MWF) ....................... 112

Bibliography 116
List of Figures

1.1 A linear array of point sensors .............................. 4
1.2 A 3 dimensional representation of a datacube. .............. 4
1.3 Element-range power distribution for pulse number 1 ....... 10
1.4 Element-range power distribution for pulse number 2000 .... 10
1.5 Range-Doppler power distribution for element number 1 ...... 12
1.6 Range-Doppler power distribution for element number 8 .... 12
1.7 Range-Doppler power distribution for element number 13 ... 13
1.8 Angle-Doppler power distribution for range bin 1 ........... 15
1.9 Angle-Doppler power distribution for range bin 50 ........... 15
1.10 Angle-Doppler power distribution for range bin 100 ......... 17
1.11 Angle-Doppler power distribution for range bin 150 ......... 17
1.12 Angle-Doppler power distribution for range bin 200 ......... 18
1.13 Angle-Doppler power distribution for range bin 225 ......... 18

2.1 A range-Doppler plot of the data-square containing obvious targets. The targets are spread over up to 15 ranges and are at Doppler bins 89, 106, 131, and 151 respectively. .............................. 34
2.2 A power profile plot of the identified targets at Doppler bins 89, 106, 131, and 151 respectively. .............................. 35
2.3 Results of using the JDL and Nonadaptive algorithms to detect an ideal target with amplitude 35 dB inserted into the ionospheric clutter region. 36
<table>
<thead>
<tr>
<th>FIGURE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.4</td>
<td>Results of using the JDL and Nonadaptive MF algorithms to detect a realistic target spread over 7 range cells and with amplitude 57 dB inserted into the ionospheric clutter region.</td>
</tr>
<tr>
<td>2.5</td>
<td>Results of using the hybrid, D$^3$, JDL, and Nonadaptive MF algorithms to detect an ideal target with amplitude 55 dB inserted into the ionospheric clutter region.</td>
</tr>
<tr>
<td>2.6</td>
<td>Results of using the hybrid, D$^3$, JDL, and Nonadaptive MF algorithms to detect a realistic target spread over 7 range cells and with amplitude 55 dB inserted into the ionospheric clutter region.</td>
</tr>
<tr>
<td>2.7</td>
<td>Results of using the RSC-PAMF, TASC-PAMF, and Nonadaptive MF algorithms to detect an ideal point target with amplitude 59 dB inserted into the ionospheric clutter region.</td>
</tr>
<tr>
<td>2.8</td>
<td>Results of using the RSC-PAMF, TASC-PAMF, and Nonadaptive MF algorithms to detect a realistic target spread over 7 range cells and with amplitude 63 dB inserted into the ionospheric clutter region.</td>
</tr>
<tr>
<td>3.1</td>
<td>A tree-like representation of the FFA method for a datacube with $M = 12$ pulses, $N = 12$ elements, spatial-partitioning-sequence=[2,2,1,3], and temporal-partitioning-sequence=[4,1,3,1].</td>
</tr>
<tr>
<td>3.2</td>
<td>A pictorial description of the Interleaved-FFA method.</td>
</tr>
<tr>
<td>4.1</td>
<td>Angle-Doppler plot of generated airborne data at range bin 223.</td>
</tr>
<tr>
<td>4.2</td>
<td>Probability of detection versus SNR for a $P_{FA} = 0.001$ in the airborne scenario using 3240 secondary samples.</td>
</tr>
<tr>
<td>4.3</td>
<td>Probability of detection versus SNR for a $P_{FA} = 0.001$ in the airborne scenario using 20 secondary samples.</td>
</tr>
<tr>
<td>4.4</td>
<td>$\Delta$MSMI versus target amplitude for the JDL, and FFA algorithms using $K=93$ secondary samples for covariance estimation.</td>
</tr>
<tr>
<td>4.5</td>
<td>$\Delta$MSMI versus target amplitude for the JDL, and FFA algorithms in the reduced sample support scenarios.</td>
</tr>
<tr>
<td>4.6</td>
<td>$\text{MSMI vs Range plots for using the AMF, and JDL methods.}$ 6998 secondary samples are used to estimate the covariance matrix.</td>
</tr>
</tbody>
</table>
4.7 MSMI vs Range plots for using the Regular FFA, Interleaved FFA, and Randomized FFA methods. 6998 secondary samples are used to estimate the covariance matrix. ........................................ 79

4.8 MSMI vs Range plots for using the AMF, JDL, and FFA methods. Only 20 secondary samples are used to estimate the covariance matrix. ............. 80

4.9 MSMI vs Range plots for using the AMF, JDL, and FFA methods. Only 20 secondary samples are used to estimate the covariance matrix. ............. 81

4.10 MSMI vs Range plots for using the Nonadaptive, and JDL methods to detect a 45dB ideal target in Ionospheric clutter region. All 4096 pulses are used in this simulation. ........................................ 83

4.11 MSMI vs Range plots for using the three FFA methods to detect a 45dB ideal target in Ionospheric clutter region. All 4096 pulses are used in this simulation. ........................................ 83

4.12 MSMI vs Range plots for using the Nonadaptive, and JDL methods to detect a 45dB ideal target in Ionospheric clutter region. Only 128 pulses are used in this simulation. ........................................ 85

4.13 MSMI vs Range plots for using the three FFA methods to detect a 45dB ideal target in Ionospheric clutter region. Only 128 pulses are used in this simulation. ........................................ 85

4.14 MSMI vs Range plots for using the Nonadaptive, and JDL methods to detect a 45dB real target in Ionospheric clutter region. All 4096 pulses are used in this simulation. ........................................ 87

4.15 MSMI vs Range plots for using the three FFA methods to detect a 45dB real target in Ionospheric clutter region. All 4096 pulses are used in this simulation. ........................................ 87

4.16 MSMI vs Range plots for using the Nonadaptive, and JDL methods to detect a 45dB real target in Ionospheric clutter region. Only 128 pulses are used in this simulation. ........................................ 88

4.17 MSMI vs Range plots for using the three FFA methods to detect a 45dB ideal target in Ionospheric clutter region. Only 128 pulses are used in this simulation. ........................................ 88
4.18 ∆ MSMI versus Sequence Ranking in the airborne scenario for a 30dB ideal target using the regular FFA method for a sample support size of 20 secondary samples. ................................. 90

4.19 Total number of unit spatial and temporal partition lengths versus Sequence Ranking in the airborne scenario for a 30dB ideal target using the regular FFA method for a sample support size of 20 secondary samples. 91

4.20 Average MSMI versus Range plot, obtained by averaging the randomized FFA detection results for a 35dB target over 10000 iterations. .............. 92

A.1 An example of a $3 \times 3$ Localized processing region in the angle-Doppler domain of the JDL method. ......................... 99

A.2 Block diagram of the unconstrained Wiener Filter. ....................... 112

A.3 Block diagram of the general sidelobe canceler. ......................... 112

A.4 Block diagram of a 4 stage unconstrained Multistage Wiener Filter. . . . . 115

A.5 Block diagram of the combined filterbank interpretation of the Multistage Wiener Filter. ........................................... 115
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACE</td>
<td>Adaptive Coherence Estimator</td>
</tr>
<tr>
<td>AMF</td>
<td>Adaptive Matched Filter</td>
</tr>
<tr>
<td>AR</td>
<td>Auto Regressive</td>
</tr>
<tr>
<td>ASD</td>
<td>Adaptive Subspace Detector</td>
</tr>
<tr>
<td>CFAR</td>
<td>Constant False Alarm Rate</td>
</tr>
<tr>
<td>CNR</td>
<td>Clutter-to-Noise Ratio</td>
</tr>
<tr>
<td>CPI</td>
<td>Coherent Processing Interval</td>
</tr>
<tr>
<td>D³</td>
<td>Direct Data Domain</td>
</tr>
<tr>
<td>DoF</td>
<td>Degrees of freedom</td>
</tr>
<tr>
<td>DRDC</td>
<td>Defense Research and Development Canada</td>
</tr>
<tr>
<td>DSTO</td>
<td>Defense Science and Technology Organization</td>
</tr>
<tr>
<td>EEZ</td>
<td>Exclusive Economic Zone</td>
</tr>
<tr>
<td>EM</td>
<td>Electromagnetic</td>
</tr>
<tr>
<td>FFA</td>
<td>Fast Fully Adaptive</td>
</tr>
<tr>
<td>FFT</td>
<td>Fast Fourier Transform</td>
</tr>
<tr>
<td>GLRT</td>
<td>Generalized Likelihood Ratio Test</td>
</tr>
<tr>
<td>GSC</td>
<td>Generalized Sidelobe Canceller</td>
</tr>
<tr>
<td>HF</td>
<td>High Frequency</td>
</tr>
<tr>
<td>HFSWR</td>
<td>High Frequency Surface Wave Radar</td>
</tr>
<tr>
<td>JDL</td>
<td>Joint Domain Localization</td>
</tr>
<tr>
<td>JNR</td>
<td>Jammer-to-Noise Ratio</td>
</tr>
<tr>
<td>km</td>
<td>kilometer</td>
</tr>
<tr>
<td>LPR</td>
<td>Local Processing Region</td>
</tr>
</tbody>
</table>
LIST OF FIGURES

MA ............. Moving Average
MSMI ........... Modified Sample Matrix Inversion
MVDR .......... Minimum Variance Distortionless Filter
MWF ............ Multistage Wiener Filter
NHD ............ Non-Homogeneity Detector
nmi ............. Nautical Mile
OTHR ........... Over the Horizon Radar
PAMF ............ Parametric Adaptive Matched Filter
PRF ............. Pulse Repetition Frequency
PRI ............. Pulse Repetition Interval
RMB ............. Reed, Mallet, and Brennan
sec .............. second
SINR ............ Signal-to-Interference-plus-Noise Ratio
SNR ............. Signal-to-Noise Ratio
STAP ............ Space-time adaptive processing
WF .............. Wiener Filter
Chapter 1

Introduction

Space-time adaptive processing (STAP) for airborne radar is a well studied area which has accumulated a substantial literature cache spanning more than 30 years. Although it was the early 1970’s that witnessed the inception of STAP through the pioneering work of Reed, Mallet, and Brennan [1], it was largely ignored for almost a decade due to the technological barriers of the time. STAP research has received a significant amount of interest recently, and, consequently, has undergone much progress.

The motivation for this thesis stems from the fact that the majority of the focus of the STAP literature has been on enhancing target detection in the airborne radar scenario. Little, if any, effort has been invested in the development of STAP techniques tailored to meet the specific needs of other interference environments. In particular, a literature survey of the available STAP techniques, indicated that no STAP method has been developed for high frequency surface wave radar (HFSWR) systems. Performance of target detection algorithms, especially non-adaptive techniques, in HFSWR, are primarily limited by sea clutter (at the near ranges) and ionospheric clutter (at the far ranges). In this regard, STAP appears to be a promising approach to deal with such interference.

This thesis has been undertaken in the context of a collaboration effort between the University of Toronto and the Defense Research and Development Canada (DRDC). The goal of this joint effort was the development of practical space-time adaptive processing schemes for HFSWR systems. In this thesis we use the operational DRDC HFSWR system as the source of the test system parameters. In addition, DRDC has provided us
1.1. MOTIVATION

with measured HFSWR data to test the algorithms developed in this work\(^1\).

A significant challenge in developing STAP for HFSWR is that traditional STAP algorithms invariably assume the interference to be spatially homogeneous, i.e., the statistics of the interference are consistent as a function of range. STAP algorithms exploit this homogeneity to estimate interference statistics for interference suppression. Specifically these statistics are estimated by averaging over the homogeneous range cells (referred to as homogenous ranges). Sea and ionospheric clutter, on the other hand, are well known to be highly non-homogeneous, requiring the development of alternative STAP approaches. However, as will be demonstrated through a preliminary analysis of the measured ionospheric clutter data sets, the fundamental limiting factor is the lack of a sufficient number of secondary range cells to estimate the interference statistics.

In this work, we evaluate the performance of several conventional STAP methods in the HFSWR scenario. We also develop a new reduced complexity multistage STAP approach, which we refer to as the Fast Fully Adaptive (FFA) approach.

1.1 Motivation

High frequency surface wave radars (HFSWR) naturally lend themselves to marine surveillance applications, and have several advantages that make them desirable for coastal surveillance applications. In particular they have large coverage area capabilities (over the horizon and beyond the exclusive economic zone (EEZ)), the lowest operating cost per unit coverage area of any radar system, as well as minimal operator intervention requirements [2]. These characteristics make HFSWR very desirable for low cost medium accuracy monitoring of low altitude aircraft and surface vessels with a coverage span as far as and beyond the EEZ. The numerous advantages associated with the HFSWR setup, and the lack of adequate and practical HFSWR interference suppression methods, motivated our research into the development of more effective means of clutter mitigation for this specific interference environment. Thus although nonadaptive and one-dimensional (space or time only) adaptive ionospheric clutter suppression algorithms

\(^1\)Another aspect of this effort was the development of an effective data model for HFSWR. However this was done outside the context of this thesis and will not be discussed here.
have been developed, they are characterized by high complexity and a large dependence on the assumed model used to describe the ionospheric clutter [3]. This thesis, therefore, focuses on the development of low complexity STAP interference cancelation techniques for HFSWR systems.

1.2 System and Data Models

In this introductory chapter, we focus on the system and data models, and present the results of some preliminary, non-adaptive, processing of HFSWR data. This chapter sets the stage for adaptive processing of HFSWR data presented in Chapter 2.

Although our primary focus will be on the HFSWR scenario, we will also evaluate the performance of the FFA algorithm in the airborne radar setup. This will allow for a fuller characterization of this new multistage adaptive processing scheme. We are therefore interested in two primary radar setups: airborne surveillance radar, and HFSWR. Both these system models will be described in the following subsections. The system and data models described here will also allow us to place the available STAP literature in context.

We begin with the airborne radar system model, which allows us to review the available STAP literature.

1.2.1 Airborne Radar System Model

Airborne radar refers to the setup where the transmit and receive antennas are mounted onto an aircraft (for example on the side of an airplane) and used to scan the ground for targets amid a background of ground and/or sea clutter that might vary rapidly. For the airborne radar scenario we will adopt the conventional STAP setup, as developed in detail by Ward in [4].

Consider a side-looking equispaced linear array\(^2\) comprised of \(N\) isotropic, point sensors separated by a distance of \(\lambda/2\), receiving an incident plane wave, as shown in Figure 1.1. Each channel receives \(M\) data samples corresponding to the \(M\) pulses in a

\(^2\)An extension of Ward’s model to include planar and 3 dimensional arrays is readily available in [5].
coherent processing interval (CPI). Thus over the span of one CPI, the $N$ channels progressively collect $NM$ space-time measurements composed of a mixture of the impinging target returns (if present) as well as background clutter, interference, and receiver noise. Upon dividing each pulse repetition interval (PRI) into $L$ separate ranges the corresponding space-time snapshots at each range are compiled into one $N \times M \times L$ data-cube (Figure 1.2). For each range bin, the received data can be stored in a length $NM$ vector $\mathbf{x}$ whose entries numbered $mN$ to $(m+1)N-1$ correspond to the returns at the $N$ elements from pulse number $m$, ($m = 0, 1, \ldots, M - 1$). This data vector is a sum of the contributions from external interference sources, thermal noise and possibly a target, and can be written as,

$$\mathbf{x} = \xi \mathbf{v} + \mathbf{c} + \mathbf{n},$$  

(1.1)
where $\mathbf{v}$ represents a normalized target signature signal, $\mathbf{c}$ the vector of interference sources and $\mathbf{n}$ the thermal noise. $\xi$ is the target amplitude and is zero under the null hypothesis (i.e., no target present).

Referring to Figure 1.2, the primary range cell refers to the range cell being tested for the presence of a target, while the secondary range cells refer to the neighboring ranges that will be used to estimate the interference covariance matrix. This interference covariance matrix will play a central role in the computation of the optimal adaptive weights in the adaptive matched filter (AMF) solution, which are used to suppress the unwanted interference and enhance the hidden target.

The vector $\mathbf{v}$ in Eqn. (1.1) is the space-time steering vector corresponding to a target at look angle $\phi_t$ and look Doppler frequency $f_t$. This steering vector can be written in terms of a spatial steering vector $\mathbf{a}(\phi_t)$ and a temporal steering vector $\mathbf{b}(f_t)$ [4],

$$\mathbf{v} = \mathbf{b}(f_t) \otimes \mathbf{a}(\phi_t),$$

$$\mathbf{a}(\phi_t) = \begin{bmatrix} 1 & e^{j2\pi f_s} & e^{j(2)2\pi f_s} & \cdots & e^{j(N-1)2\pi f_s} \end{bmatrix}^T,$$

$$\mathbf{b}(f_t) = \begin{bmatrix} 1 & e^{j2\pi f_t/f_R} & e^{j(2)2\pi f_t/f_R} & \cdots & e^{j(M-1)2\pi f_t/f_R} \end{bmatrix}^T,$$

$$f_s = \frac{d}{\lambda} \sin \phi_t,$$

where $\otimes$ represents the Kronecker product of two vectors, $^T$ represents the transpose operator, $f_s$ the normalized spatial frequency, $\lambda$ the wavelength of operation and $f_R$ the pulse repetition frequency (PRF). Eqn. (1.5) assumes the angle $\phi_t$ is measured with respect to broadside (Figure 1.1).

The term $\mathbf{c}$ in Eqn. (1.1), refers to the interference and includes the effects of clutter and possible jammers. Ground clutter, the dominant form of clutter in airborne radar, is spread across range, azimuth, and Doppler and constitutes the major source of undesirable interference in airborne radar. The Doppler spread associated with ground clutter is primarily due to the relative motion between the aircraft and the ground below. In [4],
ground clutter is modeled as the sum of the radar echoes from concentric clutter rings centered about the vertical projection of the aircraft onto the ground.

The other major source of interference in the airborne radar setup is barrage jamming, which refers to a jamming signal that is spatially correlated from element to element (i.e., behaves as a point target in space), and is temporally uncorrelated from pulse to pulse (i.e., behaves like thermal white noise temporally). A more detailed description and model of ground clutter and barrage jamming is available in Appendix A.1.

The final source of interference is the omnipresent thermal white Gaussian noise, which is uncorrelated from element to element and pulse to pulse.

The three interference sources can be lumped together into a single additive colored noise term.

Ideally a target (when present) is localized to one range bin, however in real systems the radar ambiguity function (defined formally in Eqn. (22) of [4]) leads to a spread of the target over several ranges, and this spread must be accounted for in the target model. In order to ensure an accurate estimate of the interference covariance, several (typically 2 or 3) guard ranges on either side of the primary range are used to prevent target influence from corrupting the estimate of the interference statistics. For simplicity however, we will assume that all airborne radar targets are ideal point targets, localized in range, azimuth, and Doppler.

1.2.2 HFSWR System Model

To overcome the primary source of propagation loss, namely sea surface roughness, HFSWR are operated at relatively low frequencies of the High Frequency (HF) band (between 2 and 6 MHz). One of the primary concerns with HFSWR is their high performance degradation due to highly non-homogeneous reflections from the ionospheric layer (termed ionospheric clutter). Although not the only source of unwanted signals,
the ionosphere plays a vital role in determining the performance of HFSWR based systems. However, unlike ground clutter in the airborne radar setup, ionospheric clutter in the HFSWR scenario is significantly more difficult to model. This is mainly due to the highly nonstationary characteristics of the ionosphere which vary widely with time of day, season, temperature, and location. The ionosphere is a highly ionized plasma layer of the atmosphere that spans from around 50km to about 400km above the surface of the earth. The primary source of ionization is ultraviolet radiation from the sun, which strips electrons from the neutral gas molecules leading to the formation of positive ions. The free electrons have affects on the propagation of electromagnetic (EM) radiation; in particular radio wave propagation. The level of ionization and the electron density are functions of both elevation and solar radiation intensity. As a result the electron density of the ionosphere, and consequently the characteristics of the ionospheric clutter, vary with elevation as well as time of day, season, and sun-spot activity.

There are two available data models for ionospheric clutter [3,6]. The work of Ravan et al. detailed in [7] indicated that the model proposed by Fabrizio [3], although analytically appealing, can not be used to model the measured ionospheric data acquired by the DRDC HFSWR. A more accurate model is developed by Riddolls [6, 8] which extends the model introduced by Coleman [9,10], by simultaneously accounting for group delay, direction of arrival, location, and Doppler. The author uses a ray tracing model and treats irregularities as perturbations of a “quiescent” path solution without irregularities. This model has been recently implemented by Ravan [7].

Although ionospheric clutter is the dominant clutter source in the far ranges, it is sea clutter, and in particular the Bragg component of the sea clutter that dominates the near ranges. Sea clutter is characterized by a nonzero Doppler signature that is spread over the frequencies in the vicinity of the zero Doppler frequency; it exhibits particularly
strong returns at the Bragg Doppler frequencies given by,

\[ f_{\text{Bragg}} = \pm \sqrt{\frac{g}{\pi \lambda}} \]  

(1.6)

where \( g = 9.8 \text{m/s} \) is the gravitational acceleration, and \( \lambda \) is the wavelength of the radar signal. The Bragg component is due to the constructive interference of radar echoes from ocean waves with a wavelength equal to \( \lambda/2 \) [11]. It should be noted that this Bragg clutter line begins to recede with increasing range, and almost disappears completely for ranges beyond \( 200 \text{km} \) [12].

Having briefly introduced the types of clutter encountered in HFSWR, we next turn our attention to the target model used in this setup. Ideal targets in the HFSWR scenario can be modeled (quite accurately) in a fashion similar to targets in airborne applications (i.e., point sources localized in azimuth, Doppler, and range), and as a result we will use the conventional space-time steering vector, developed in the airborne radar literature [4], and summarized in Section 1.2.1, to model target returns in HFSWR. However in order to account for the spread of the target in range we will also use a realistic target space-time vector which is spread over several range bins, and extracted from measured high signal-to-noise-ratio (SNR) target returns present in separately measured HFSWR data cubes. We will have more to say on this subject later on.

### 1.3 Preliminary Data Analysis: Measured Data

This section details the non-adaptive data processing conducted on the data sets provided by DRDC, and used in our evaluation of several reduced complexity STAP techniques in the HFSWR scenario. In particular we will restrict our discussion in this section, to the analysis of the data set `hfswr_data_25mar2002_030257.mat`.

We will begin by briefly describing the measurement system used to collect the data
sets used for the HFSWR simulations in this work. The reader is referred to [12] for a
detailed description of the measurement setup. The measured ionospheric clutter was
obtained above Cape Race in Newfoundland located on Canada’s East coast using a HF-
SWR system operated by DRDC. The radar operates at a carrier frequency of 3.1 MHz.
The data was obtained using a 16 element linear array with inter-element separation
of 33.33m (corresponding to 0.344\lambda), and gathered over a time period of 4.37 minutes,
divided into 4096 pulses with an effective pulse repetition frequency (PRF) of 15.625Hz
(setting the maximum resolvable Doppler frequency to \pm 7.8125Hz). The range span was
about 400km divided into 270 consecutive range cells each covering a span of 1.5km. It
should be noted that the measured data we received was oversampled in range by a fac-
tor of approximately 4, which means that consecutive ranges were correlated to a certain
degree. This correlation in range has a negative impact on the accuracy of the estimate
of the interference covariance in the AMF solution, which requires that the secondary
ranges used in the covariance estimate to be independent.

The data analysis conducted in this section focuses on non-adaptive processing, and is
used to characterize the measured data set. Some sample results are provided illustrating
the interference distribution as a function of angle, Doppler and range.

1.3.1 Element-Range Plots

The first set of results focus on the range dependence of the interference. Figure 1.3 plots
the power distribution, in dB, of the first pulse as a function of element and range. There
appears to be three distinct segments in range - a near range segment extending up to
approximately range cell 50 with significant interference power, a segment between range
cell 50 and 180 with lower power, and finally, the range cells dominated by ionospheric
clutter past range cell 180. Figure 1.4, corresponding to pulse number 2000, reconfirms
this impression.

The figures also point to a problem with element 13. This element appears to be
1.3. PRELIMINARY DATA ANALYSIS: MEASURED DATA

Figure 1.3: Element-range power distribution for pulse number 1

Figure 1.4: Element-range power distribution for pulse number 2000
1.3. PRELIMINARY DATA ANALYSIS: MEASURED DATA

1.3.1 Attenuation

This same problem exists in all the data-sets given to us by the DRDC. However, interestingly, focusing on this one element suggests that this receiver is not totally “dead”. The associated angle-Doppler plot in Fig. 1.7, presented later, has the appropriate characteristics, only they are attenuated by as much as 40dB.

1.3.2 Range-Doppler Plots

This section focuses on range-Doppler plots for individual elements. The Doppler domain is ‘accessed’ by performing a Fourier transform on the available pulses at each range for a specific element. Figures 1.5 and 1.6 plot the power distribution of the radar signal returns as a function of range and Doppler. As is clearly seen, there are three distinct clutter regions; the near region extending to approximately 140km (comprising approximately the first 50 range bins) dominated by the Bragg lines and sea clutter, a region at the far ranges beyond 330km dominated by the ionospheric clutter and, interestingly, a middle region of sea clutter with less structure. This corroborates with our initial impression from Figs. 1.3 and 1.4. As we will see in later plots, this variation in clutter structure is also visible in the angle-Doppler plots presented in Section 1.3.3 and has significant implications for interference suppression.

In both plots the Bragg lines are clearly visible. The advancing and receding lines are at $\pm 0.18 \, Hz$ corresponding to the expected Doppler frequency given by Eqn.( 1.6). An interesting conclusion can be drawn by plotting the range-Doppler power distribution for element 13 (Fig. 1.7). While our initial understanding was that channel 13 was “dead”, Fig. 1.7 reveals that in fact this channel appears to be receiving data, but attenuated by as much as 40dB with respect to the other elements.

These range-Doppler plots provide crucial information for the STAP process. As we will see in Chapter 2, a fundamental limitation of STAP is the need for training data to estimate the statistics of the interference within a primary range cell. This training data, clearly, must be statistically homogeneous with respect to the range cell under test. The
1.3. PRELIMINARY DATA ANALYSIS: MEASURED DATA

Figure 1.5: Range-Doppler power distribution for element number 1

Figure 1.6: Range-Doppler power distribution for element number 8
Figure 1.7: Range-Doppler power distribution for element number 13
range-Doppler plots indicate that there is limited training available within each clutter region. This is independent of whether the clutter is actually homogeneous within each of the three clutter regions described above.

### 1.3.3 Angle Doppler Plots

In this subsection we will analyze the angle-Doppler plots for individual range cells. These plots are useful since they illustrate the power distribution in angle-Doppler space; the two Fourier spaces corresponding to the spatial and temporal domains in which STAP will be implemented. It is well accepted that it is easier to suppress localized interference (localized near a specific Doppler/angle). We will see that the range cells dominated by the Bragg lines and those dominated by the ionospheric clutter appear to have a more coherent structure.

Figures 1.8 and 1.9 plot the angle-Doppler structure for range bins 1 and 50 respectively (ranges of 62.75km and 136.25km respectively). As is clear from Fig. 1.8 the interference in the first range bin has a clear structure - the Bragg lines near zero Doppler are visible across all angles. Also, the interference is localized to a few angles. We are also able to identify two major sources of discrete interference: (i) a high power interference source spread over all range and Doppler, and localized in azimuth (24°-46°). This source has a coherent structure and we therefore believe it to be a communication channel; (ii) a high power interference source spread over all ranges, and localized in Doppler (4Hz and 2Hz) and azimuth (60°-90°), including two relatively strong source at approximately 35° (relative to broadside). As we will see, these two interference sources appear in all range cells.

Figure 1.9 plots the power distribution in angle and Doppler for range cell 50, at the edge of the first interference region. Comparing this plot to Fig. 1.8 one can visualize the loss of structure in the interference. The interference here is far more spread out. This effect is further illustrated in Figs. 1.10 and 1.11 corresponding to range bins 100 and
1.3. PRELIMINARY DATA ANALYSIS: MEASURED DATA

Figure 1.8: Angle-Doppler power distribution for range bin 1

Figure 1.9: Angle-Doppler power distribution for range bin 50
In these two figures the lack of structure in the clutter is particularly striking. This has significant implications for the STAP process; with its focus on a localized region in angle-Doppler space, an algorithm such as the joint domain localized (JDL) processing scheme [13, 14] may be useful here.

In the range cells dominated by ionospheric clutter, a structure is again visible in the angle-Doppler space. Figures 1.12 and 1.13 illustrates the strong ionospheric interference close to zero-Doppler and the external interference is again clearly visible.

In summary, the angle-Doppler plots suggest both the potential of STAP to suppress interference and also some cautionary tales. As our preliminary analysis of the HFSWR data revealed, the data cube appears to have three distinct regions. In the first region, dominated by the Bragg lines, a clear structure is visible and a “traditional” adaptive algorithm such as the sidelobe canceler followed/preceded by Doppler processing may be adequate. In the third region, dominated by ionospheric clutter, STAP is required due to the inherent non-stationarity of the clutter. In the middle region, again dominated by sea clutter, no structure is visible and a JDL-based algorithm might be required.

1.4 Thesis Overview

The remainder of this thesis is organized as follows; Chapter 2 reviews STAP and several STAP algorithms and presents simulation results that are used to evaluate the performance of these methods in HFSWR systems. Chapter 3 introduces the Fast Fully Adaptive approach as well as three different variants. The fast adaptive matched filter is also developed in this chapter by integrating the FFA approach into the optimal adaptive matched filter (AMF). Chapter 4 presents several simulation results of applying the FFA algorithm to detect ideal and realistic targets in the airborne and HFSWR setups. Chapter 5 wraps up the thesis with some conclusions and several suggestions for future research in the context of the FFA and STAP for high frequency (HF) radar systems.
1.4. THESIS OVERVIEW

Figure 1.10: Angle-Doppler power distribution for range bin 100

Figure 1.11: Angle-Doppler power distribution for range bin 150
1.4. THESIS OVERVIEW

Figure 1.12: Angle-Doppler power distribution for range bin 200

Figure 1.13: Angle-Doppler power distribution for range bin 225
Chapter 2

Processing of HFSWR Returns

The central goal of this thesis is the development and analysis of practical STAP schemes for HFSWR radar systems. Our preliminary analysis of the dataset suggested that the most important issue with the available data is (a) the non-stationarity of the clutter and (b) the lack of sufficient secondary data to estimate the space-time covariance matrix. In this regard, of special interest are adaptive schemes that address clutter non-homogeneity and algorithms with fewer degrees of freedom. Examples of such algorithms include the Joint Domain Localized (JDL) algorithm [14], the direct data domain ($D^3$) algorithm [15] and the combination of the two into the hybrid algorithm [16]. Other algorithms of interest are the parametric adaptive matched filter (PAMF) [17], and multistage Wiener filter (MWF) [18,19]. The $D^3$ and hybrid algorithms were designed specifically for non-homogeneous clutter, while the JDL, PAMF, and MWF were designed to address the issue of limited sample support.

As an initial step in this direction, we will begin by reviewing the basic concepts of STAP through a description of the optimal fully adaptive processor (also referred to as the adaptive matched filter (AMF)). Next we will briefly review some of the available adaptive processing techniques used in HFSWR. We will then turn our attention to the primary subject of this Chapter; applying low-complexity STAP to the HFSWR setup.
2.1 REVIEW OF FULLY ADAPTIVE STAP

Particularly we will review each of the JDL, D3, hybrid, PAMF, and MWF algorithms, and discuss the various advantages and drawbacks associated with each method, as well as how each of these schemes fits into the HFSWR setup. Finally, we will present several simulation results that will allow us to examine the effectiveness of STAP techniques in suppressing ionospheric clutter.

2.1 Review of Fully Adaptive STAP

In this section we will review fully adaptive STAP, and describe the advantages and drawbacks associated with the optimal solution. This optimal processor is of particular interest to us as it will form the main building block of the fast fully adaptive method we will develop in Chapter 3. Since STAP is strongly rooted in airborne surveillance radar, we will use the airborne system model, described in Section 1.2.1, to present the STAP techniques of interest to us in this work.

In this section we are concerned with the linearly constrained minimum variance distortionless response (MVDR) Wiener filter (WF), also referred to as the the matched filter (MF). The received signal vector can be written as,

\[ x = \xi v + n \]  \hspace{1cm} (2.1)

where \( \xi \) is the target amplitude, \( v \) is the target space-time steering vector at the angle and Doppler of interest, and \( n \) is the total colored interference.

STAP applies a set of weights, \( w \), to this received data vector to form a statistic,

\[ y = |w^H x| \]  \hspace{1cm} (2.2)

where a target is declared present if \( y \) is greater than a chosen threshold, and absent otherwise. The threshold, choice of weights, and interference environment determine the
probabilities of detection and false alarm.

Assuming a data model where all the interference signals are zero mean, jointly stationary, complex Gaussian processes, the MVDR-WF, is known to be the optimal adaptive processor under the maximum signal to interference plus noise ratio (SINR) criterion. Under the Gaussian assumption the maximization of the probability of detection can be shown to be equivalent to the maximization of the SINR \cite{20}. Thus under the assumption of \textit{a priori} knowledge of the interference statistics, and as a result of the error covariance matrix \( \textbf{R} \), the matched filter solution leads to the computation of the following optimal weight vector,

\[
\textbf{w}_{\text{opt}} = \textbf{R}^{-1}\textbf{v},
\]

where \( \textbf{R} = \mathbb{E}\{\textbf{nn}^H\} \) is the known interference covariance matrix. This is also the matched filter solution under colored noise.

Unfortunately, it is seldom the case that the interference statistics are known priori, and thus must be estimated from the neighboring ranges. This leads to an alternate suboptimal solution called the Adaptive Matched Filter (AMF), or the fully adaptive processor, which is identical to the WF with the known covariance \( \textbf{R} \) replaced by its maximum likelihood estimate \( \hat{\textbf{R}} \) given by,

\[
\hat{\textbf{R}} = \frac{1}{K} \sum_{r=1}^{K} \textbf{x}_r \textbf{x}_r^H
\]

where \( \textbf{x}_r \) is the space-time data snapshot at the \( k^{th} \) secondary range cell. We can then evaluate the following modified sample matrix inversion (MSMI) statistic \cite{21},

\[
\Lambda_{\text{AMF}} = \frac{\|	extbf{v}^H \hat{\textbf{R}}^{-1} \textbf{x}\|^2}{\textbf{v}^H \hat{\textbf{R}}^{-1} \textbf{v}}.
\]

In Gaussian interference, this statistic has the centrally useful property of constant false
alarm rate (CFAR) independent of $R$. The false alarm rate is a function of $N$, $M$, and $K$ only.

Clearly the fully adaptive processor assigns an adaptive weight to each of the $NM$ degrees of freedom (DoF). However for the estimate $\hat{\mathbf{R}}$, of the interference covariance matrix to be accurate, the number of *statistically homogeneous* range cells used to produce this estimate must be at least $2NM$ according to the Reed-Mallet-Brennan (RMB) rule [1]. Unfortunately such generous sample support is rarely available in practice (and is even more rare in the HFSWR scenario), leading to the computation of an inaccurate estimate of the true interference statistics. As seen in Chapter 1, while $N = 16$, and $M = 4096$, we only have 93 range cells ($\ll$ than the $2NM = 131072$ secondary ranges required by the RMB rule) within the ionospheric region, a quarter of which are redundant due to the oversampling of the measured data in range. It is, therefore, the requirement of statistically homogeneous sample support that is fundamentally impossible to meet in practice. In addition, the computation of the optimal weight vector, involves the evaluation of the matrix inverse of $\hat{\mathbf{R}}$, which leads to an impractical computational load (in the order of $(NM)^3$) for most practical choices of $M$ and $N$.

### 2.2 Literature Review of HFSWR Processing Techniques

This section reviews some of the previously published work in adaptive processing for HFSWR. Largely the work appears to have focused on the reduction of the degrees of freedom (DoF) in the processing scheme so as to reduce computational load and the requirements of statistically stationary sample support.

Some of the early works by researchers at DRDC include [22, 23] which discuss the use of horizontal dipoles as auxiliary antennas to take on the role of sidelobe cancelers.

---

1 A sidelobe in antenna terminology is the response pattern in a direction other than the look direction.
used in the suppression of skywave interference. The author derives the adaptive weights to optimally suppress interference. A reference signal is added to obtain the weights using a Wiener solution. In [24] the author extends the results [22, 23] to include a comparison between the use of horizontal and vertical dipoles for interference suppression. The author suggests the use of horizontal polarization to cancel interference close to the target location (whose signal is vertically polarized). This work may not have significant impact on the present project since all the antennas, used to obtain the measured data, use the same (vertical) polarization. Furthermore, these works adaptively process the space dimension only.

Other efforts within the DRDC include the coherent sidelobe canceler of [25], where the authors investigate the optimal ordering of Doppler processing, beamforming, and interference cancelation. One unfortunate result is that the optimal ordering is interference dependent, and consequently time dependent, since the interference sources in HFSWR demonstrate a large variability with time. A similar result was demonstrated in the context of airborne radar in [13]. In [25], the sidelobe canceler is shown to be effective against a single spatially confined source. An interesting, and sobering, contribution in this paper is the demonstration of the wide spatial and temporal variation in the ionospheric clutter characteristics.

These works would, we believe, be considered “classical” in the space-time adaptive processing community, and underline the complexity of the problem being tackled in this project.

A significant fraction of the work in adaptive processing for HFSWR is apparently led by Dr. Yuri Abramovich and Dr. Giuseppe Fabrizio at Defense Science and Technology Organization (DSTO) in Australia [3, 26–30]. In particular, Fabrizio has several contributions developed in detail in his thesis and then several later works that are reviewed.
below. The work of Abramovich focuses largely on the underlying phenomenology and measurements [31].

The work of the group led by Fabrizio [3, 26–30] has focused on the development of the adaptive coherence estimator (ACE) and its variant, the spatial adaptive subspace detector (ASD) [26], for over-the-horizon (OTH) radar systems. In the following we largely focus on their most recent contributions [28–30] presented as an improvement on their previous work [3, 26, 27].

The ACE test, like the modified sample matrix inversion (MSMI) statistic, has the important CFAR property. However ACE has the CFAR property even when the data within the range cell under test has a different scale from that in the secondary data. Unfortunately, ACE is highly susceptible to target mismatch and hence the motivation for the development of the ASD detector.

The ASD detector treats a single target as a signal subspace with rank greater than 1 [29]. In their latest work [30], Fabrizio et al. address the issue of unwanted signals in the primary data. They propose a generalized likelihood ratio test (GLRT) to address this issue. To address the issue of target mismatch, they model both the target and the discrete interference source within the primary range cell as low-rank sources given by a linear combination of closely spaced Doppler frequencies. The GLRT is then formed by maximizing over the parameters of both the target and discrete interference. Unfortunately, a significant drawback is that the implementation of the GLRT requires exact knowledge of the parameters of the interference. This necessitates a two-pass approach wherein the detector acquires some knowledge of the existence of the interference and its parameters in real time. This two-pass approach is reminiscent of the two-pass approach proposed by Adve et al. in [32, 33]. It should be noted that other than the experimental results, the formulation in [30] does not specifically target HFSWR.

The work in [30] focuses on adaptive Doppler processing exclusively. Previously the
authors developed a STAP algorithm \cite{28} for OTH radar using beamspace-range\textsuperscript{2} processing. This paper is of interest because the authors claim that with \(N\) spatial elements and \(M\) time taps, they can reduce the dimension of the STAP filter to \((M + N)\) as opposed to the usual \(MN\). The move to beamspace is similar to the JDL scheme’s approach of working in azimuth-Doppler space, in that the spacing between beams is not restricted to the use of a FFT. Interestingly, the authors use \textit{fast time} samples\textsuperscript{3} pulses to form the Doppler steering vector, \textit{as opposed to the traditional slow time samples} - hence the name beamspace-range processing.

The other significant group of researchers working in adaptive processing for HF radar is in China. In \cite{34} the authors propose a scheme for clutter mitigation that both varies the clutter weights to counter non-stationary ionospheric clutter within a coherent processing interval (CPI) while maintaining some stability in the gain on target. Two issues with this paper appear to be the need for partitioning the overall CPI into sub-CPIs, increasing the computation load, and the somewhat ad-hoc nature of the proposed processing scheme with transitions from sub-CPI to sub-CPI.

The work in \cite{35} is the basis for experimental results reported in \cite{36}. In \cite{35} the authors claim that clutter and target returns only impact the positive frequencies; and not the negative frequencies, however the basis for this claim is not clear to us.

In \cite{37} the authors extend temporal only processing to the use of STAP. However, the approach used is the classical STAP (elaborated on in the previous section) approach of estimating the covariance using secondary data without any specific attention paid to homogeneity.

\textsuperscript{2}For the special case of a uniform linear array, the beamspace domain can be accessed through a Fourier transform of the spatial domain.

\textsuperscript{3}The fast time samples refer to the range samples corresponding to a certain pulse, while the slow time samples refer to the sequence of returns across all the pulses of a coherent processing interval corresponding to a certain range.
2.3 Review of Low Complexity STAP Techniques

In Section 2.1 we reviewed the fully adaptive processor. Although it provides the optimal solution, it is highly impractical due to its large sample support requirements, and its large computational load. To address these issues researchers have developed techniques that are characterized by a lower-complexity, and that utilize fewer adaptive DoF. Some popular low complexity approaches, developed in the context of airborne radar, include, the Joint Domain Localized (JDL) algorithm [14,38], the Parametric Adaptive Matched Filter (PAMF) [17] and the reduced rank Multistage Wiener Filter(MWF) [18, 19]. In the following sections we will briefly describe several such low-complexity approaches. The details of the algorithms are left to Appendix A.2.

2.3.1 Joint Domain Localized (JDL) Processing

One popular low complexity STAP scheme is the JDL algorithm which was first proposed in [13] with strict constraints on the algorithm and generalized in [14] to eliminate these constraints and allow for the use of real-world antenna arrays with mutual coupling. The JDL processor adaptively processes the received data in the angle-Doppler domain, i.e., after transforming the received space-time data to the angle-Doppler domain. This transformation is accomplished by an inner product with a set of space-time steering vectors, i.e., the data at azimuth angle $\phi$ and Doppler frequency $f$ is given by

$$\hat{x} = v^H(\phi, f)x,$$

(2.6)

where $x$ is the measured space-time data at a specific range cell, and $\hat{x}$ represents angle-Doppler data. Choosing a set of $\eta_a$ angle and $\eta_d$ Doppler bins centered at the look angle-Doppler forms the data set, in the angle-Doppler domain, used for adaptive processing. This set of $\eta_a$ angle and $\eta_d$ Doppler bins is said to form the Localized Processing Region.
2.3. REVIEW OF LOW COMPLEXITY STAP TECHNIQUES

Unfortunately, there is no known criteria for optimally choosing the spacings in angle and Doppler; in most implementations the optimal spacings have been found empirically [7]. The primary advantage of the JDL approach is the fact that only $\eta_a$ angle and $\eta_d$ Doppler bins are used in the adaptive process with values in the range of 3 - 11 each. Significantly reducing the adaptive DoF in this manner yields corresponding reductions in required sample support and computation load. Appendix A.2.1 describes the JDL algorithm in more detail.

Despite the benefits of working in the angle-Doppler domain, the performance of the JDL algorithm, like all statistical algorithms, depends on the accuracy of an estimated interference covariance matrix. This in turn places homogeneity constraints on the interference. Unfortunately this constraint is often violated in practice, and more so in the case of high-power clutter returns from the ionospheric layer, which are characterized by their inherent inhomogeneity, presence of discrete-like clutter sources, and variability. However, if the various parameters that characterize the ionospheric clutter have a coherence time longer than several CPIs, and are coupled with the use of algorithms such as JDL, the detrimental effect of the inhomogeneities may be minimized [39].

The essential cause of this problem is the dependence of the algorithm on the statistics of the interference. Alternative, deterministic, adaptive approaches to suppress inhomogeneous clutter, have been developed to solve this problem. One notable class of deterministic algorithms are the so-called direct data domain ($D^3$) algorithms. Examples of this class of algorithms are the one-dimensional $D^3$ algorithm, as developed in [16, 40], and the two-dimensional ($D^3$) of [41]. We will describe the one-dimensional $D^3$ algorithm in the next subsection.

2.3.2 $D^3$ and Hybrid Approaches

The $D^3$ algorithm, developed in [16] and improved upon in [40], is a purely non-statistical algorithm which does all its processing using only the primary range cell without relying
on the neighboring ranges to extract an estimate of the interference covariance matrix. As a result the D³ algorithm is immune to both clutter non-homogeneities in the secondary data and target-like, discrete, interference sources within the primary range cell. The discrete interference may occupy multiple angle and Doppler bins. This scenario is more relevant to us since the ionosphere, with its highly nonhomogeneous nature, is expected to resemble, to a certain degree, a source of high power discrete interference spread over numerous range cells leading to a contamination of the homogeneity of the secondary data.

The D³ algorithm computes an adaptive weight vector that does not rely on the computation of an error covariance matrix. It first forms a temporal residual matrix, \( \mathbf{A}_t \), free from any target returns by subtracting the returns from all the consecutive pulses after appropriate scaling. It then selects a temporal weight vector that maximizes the signal-to-interference-plus-noise ratio (SINR):

\[
\hat{\mathbf{w}}_t = \arg \max_{\mathbf{w}_t} \frac{|\mathbf{w}_t^H \mathbf{b}(f_t)|^2}{\|\mathbf{A}_t^H \mathbf{b}(f_t)\|^2}.
\]

The solution results from solving a generalized eigenvalue problem [40]. After forming a similar spatial weight vector, \( \hat{\mathbf{w}}_s \) using the data from the spatial domain, the final space-time weight vector is given by \( \mathbf{w} = \hat{\mathbf{w}}_t \otimes \hat{\mathbf{w}}_s \). A more rigorous description of the one-dimensional D³ algorithm is available in Appendix A.2.2.

In the DRDC data sets, \( M = 4096 \), making the matrix in question singular. Even finding the generalized eigenvalues becomes difficult since the algorithms involved become unstable. However this problem can be overcome by adding a small diagonal loading to the matrix.

The drawback of the D³ approach is that by focusing exclusively on the data within a single range cell, it completely ignores all useful correlation information. Since ionospheric clutter is a mix of correlated and discrete interference, the performance of the D³
2.3. REVIEW OF LOW COMPLEXITY STAP TECHNIQUES

The D³ algorithm is inadequate. One other disadvantage of the D³ algorithm is that it requires the computation of the inverse of an \( (M - 1) \times (M - 1) \) matrix which can be quite a daunting task when the number of pulses is large. To overcome some of the weaknesses of the D³ method, the hybrid algorithm was developed in [16]. The hybrid is a two-stage algorithm whose building blocks are the D³ and JDL algorithms, respectively. It uses the D³ weights as an adaptive transform to the angle-Doppler domain. The angle-Doppler data for angle \( \phi \) and Doppler \( f \) is given by

\[
\hat{x} = w^H(\phi, f)x, \tag{2.8}
\]

where \( w(\phi, f) \) is the D³ adaptive weight vector corresponding to angle \( \phi \) and Doppler \( f \). Choosing \( \eta_a \) angle and \( \eta_d \) Doppler bins forms the LPR for the JDL algorithm. During the statistical adaptive processing stage, the JDL nulls the correlated interference by computing a weight vector derived from the estimate of the error covariance matrix in the transformed domain. Thus the hybrid algorithm is capable of suppressing both uncorrelated and correlated interference in the target range, and as a result theoretically performs better than both the JDL and D³ algorithms. We provide a more detailed description of the hybrid algorithm in Appendix A.2.3.

2.3.3 Parametric Adaptive Matched Filter (PAMF)

Another reduced-domain algorithm that shows robustness to the presence of heterogeneous interference in the target range, is the Parametric Adaptive Matched Filter (PAMF) algorithm [17]. The underlying principle is the use of linear estimation theory to obtain an estimate of the inverse of the error covariance matrix, but using fewer computation cycles than the conventional AMF method. PAMF is based on the block-LDL decomposition of the covariance matrix, \( R \), where the non-zero block elements of the \( p \)-th row of the lower triangular block matrix, \( L \), are the coefficients of a multichannel
autoregressive (AR) process of order $p$, whose error covariance matrix is the $p$-th block matrix of the block diagonal matrix $D$. Since $R$ is not known \textit{a priori}, the coefficients of $L$ and $D$ must be approximated by assuming the underlying process is a $P$-th order multichannel AR process and using an appropriate parameter estimation algorithm to estimate the coefficients of this multichannel AR($P$) process. An excellent discussion of multichannel AR processes and several corresponding parameter estimation techniques are available in [42]. A more detailed description of the PAMF algorithm is available in Appendix A.2.4.

It is claimed in [17] that the PAMF method also shows robustness to the presence of discrete interference in the secondary data and has performance that rivals that of the MF even in the case of limited sample support. Unfortunately the PAMF method is not CFAR (i.e., the false alarm rate varies with the interference level), however simulation results suggest that for certain PAMF configurations CFAR like behavior does occur [17].

The main drawback of the PAMF is difficulty associated with the selection of an optimal AR order and the complexity of its implementation. Although the authors in [17] claim that an AR order of 3 is sufficient to accurately model the underlying process in airborne radar, no such investigation has been conducted in the HFSWR case which may very well require a much higher AR order given the highly nonhomogeneous nature of the ionospheric clutter. Another issue with the PAMF is its larger implementation complexity as compared to other available low complexity methods such as the JDL and $D^3$. A final drawback of using the PAMF is that it lacks the CFAR property. Although simulation results in [17] indicate that for certain configurations the PAMF does yield CFAR like behavior, this is not a general rule of this algorithm and as a result reduces the practicality of the PAMF to a certain extent.
2.3.4 Multistage Wiener Filter (MWF)

The multistage Wiener filter (MWF) developed in [18,19] is an equivalent representation of the optimal Wiener filter that admits a recursive decomposition into a series of scalar Wiener. We defer the description of the MWF to Appendix A.2.5. This decomposition eliminates the need of directly working with the full $MN \times MN$ covariance matrix, and instead only deals with the computation of numerous scalar Wiener Filter weights. Although this might seem as a huge advantage, especially for large values of the product $MN$, the complexity reduction gained by eliminating the need for computing the inverse of a large dimensional matrix, is regained through the large number of matrix products and error signal recombinations constituting the analysis and synthesis stages of the MWF. One solution to this complexity problem comes in the form of the reduced rank MWF where the decomposition is interrupted at the $P^{th}$ stage [18]. This can be achieved by discarding the last $(NM - 1 - P)$ rows of the analysis filterbank matrix, $L$. One justification for this is that the covariance of the observed data tends to become white as the number of stages increases, and therefore a close approximation to the optimal Wiener filter can be obtained by truncating the $P$ ‘deepest’ stages where the vector $h_k$ approaches 0 [18]. However this reduced rank solution might still be impractical in the HFSWR case where the product $MN$ is extremely large. Another solution to the large complexity associated with the MWF, and without reverting to a reduced rank filter, can be obtained by incorporating the conjugate gradient method into the MWF [43].

2.4 Applying STAP to HFSWR

In the previous sections we reviewed the optimal fully adaptive filter and several sub-optimal, reduced complexity STAP methods. All the described methods can be directly applied to the HFSWR setup under the assumption of an interference environment whose
statistics are sufficiently homogeneous across range so as to allow for an acceptable estimate of the covariance to be obtained. The assumption of a Gaussian interference environment would also allow retaining the CFAR properties of the MSMI statistic used in the AMF, JDL, hybrid, and MWF methods. Unfortunately the ionospheric clutter is known to be a form of highly inhomogeneous and nonstationary interference. However over shorter periods of time, we hope that the ionospheric clutter maintains enough homogeneity over the secondary ranges used in the covariance estimation, so as to render the STAP methods, discussed in the previous section, useful. We will therefore assume a sufficiently stationary and homogeneous environment with complex Gaussian interference statistics, and use the results of the STAP simulations to verify the validity of this simplifying assumption.

To perform the simulations, we use the measured dataset, described in Section 1.3. Our primary focus will be the assessment of how effectively STAP can mitigate ionospheric clutter. As a result we will mainly focus on the ranges dominated by ionospheric clutter, namely range cells 176 to 270. The fully adaptive processor, although optimal, is inapplicable to our test setup. The reason for this is two-fold; first the minimum required sample support, as dictated by the RMB rule, is $K = 2MN = 2(4096)(16) = 131072$ secondary samples, and only 94 ionospheric clutter range cells are available; second the computational load associated with estimating and inverting the $65536 \times 65536$ covariance matrix is beyond the capabilities of most modern processors. We must therefore direct our efforts towards the more practical, albeit suboptimal, approaches suggested in the previous section.
2.5 Performance of Available STAP Algorithms in HFSWR

In this section we will attempt to evaluate the performance of the existing low-complexity STAP methods, described in Section 2.3, in the HFSWR setup. The simulation results to be presented in this section, will allow us to assess the level of utility of STAP in HFSWR applications.

We will begin by describing the realistic target model which takes the radar ambiguity function into account, and which will be used in addition to the ideal target model, described in Sections 1.2.1, and 1.2.2, to characterize the performance of the chosen algorithms. Next we will present several simulation results detailing the performance of these algorithms, followed by a ‘Discussions’ section where we will identify the implications of the simulation results.

2.5.1 Realistic Target Model

Assuming a point target model has several advantages such as simplicity, localization to a single range bin, and the elimination of the need to incorporate the ambiguity function into the target model. It also does have a role in characterizing the performance of the algorithm being tested. Unfortunately ideal targets do not exist in the real world and we must therefore choose a more realistic target model. To account for the spread of real targets in range we used obvious realistic targets directly extracted from a sample data-square (since only one spatial channel was used) provided to us by DRDC. Figure 2.1 shows a range-Doppler plot of this data-square where the targets are indicated by arrows. Note that the ideal target model does have a role to play in identifying the performance of the algorithms under test. Figure 2.2 shows several superposed cross-sections of the range-Doppler plot at the Doppler bins corresponding to some of the identified targets.
2.5. PERFORMANCE OF AVAILABLE STAP ALGORITHMS IN HFSWR

Figure 2.1: A range-Doppler plot of the data-square containing obvious targets. The targets are spread over up to 15 ranges and are at Doppler bins 89, 106, 131, and 151 respectively. It should be noted that the range span of the data-square places it in the region dominated by Bragg lines and thus little or no ionospheric clutter masks the targets.

As can be seen from the plots in Figure 2.2 the realistic targets exhibit a range-power profile that extends over several range bins with the peak power located near the center of the target’s range span. In our simulations we will particularly focus on the targets located at Doppler $-0.1526$Hz (Doppler-bin 89) and $-0.087$Hz (Doppler-bin 106) respectively. It should be noted that we first normalized the targets by dividing by the magnitude of the peak target amplitude, and then dividing by the magnitude of the corresponding steering vector, before actually injecting the target at the desired angle and Doppler locations. A final yet important detail to note is that the magnitude of both the ideal and realistic targets used in the simulations are an absolute magnitude and not a signal-to-noise ratio (SNR).
2.5. PERFORMANCE OF AVAILABLE STAP ALGORITHMS IN HFSWR

Figure 2.2: A power profile plot of the identified targets at Doppler bins 89, 106, 131, and 151 respectively.

2.5.2 Simulation Results

In this section we provide some of the results obtained by using the algorithms described in the previous sections to detect ideal and realistic targets injected into the region dominated by ionospheric clutter. Both the ideal and realistic targets used were injected at an angle of 35° and a Doppler of 0.18Hz. The ideal target used is localized to a single range bin, while the realistic target used is that extracted from the data-square at Doppler bin 89, and is spread over ranges approximately 7 ranges.

To assess the performance of the algorithms under investigation we focus on the peak MSMI at the target range and the difference in MSMI (in dB) between the peak MSMI and the second highest statistic, denoted as $\Delta$MSMI. The larger the $\Delta$MSMI the larger the capability of an adaptive algorithm to distinguish between a target and residual interference, and thus the better its performance. We begin by presenting some of the JDL algorithm results.

Figure 2.3 plots the MSMI statistic versus range, at the target Doppler, for an injected
2.5. PERFORMANCE OF AVAILABLE STAP ALGORITHMS IN HFSWR

Figure 2.3: Results of using the JDL and Nonadaptive algorithms to detect an ideal target with amplitude 35 dB inserted into the ionospheric clutter region.

Figure 2.4: Results of using the JDL and Nonadaptive MF algorithms to detect a realistic target spread over 7 range cells and with amplitude 57 dB inserted into the ionospheric clutter region.
ideal target using JDL. In this figure an absolute target amplitude (not SNR) of 35dB is used. Figure 2.4 shows an MSMI vs range plot for the injected realistic target of amplitude 57dB, using JDL. These target amplitudes appear to be large, but in fact, as shown in the figures, they cannot be detected using non-adaptive processing. The results presented in these figures for JDL are for an optimal angle and Doppler spacing obtained via a brute force search using a LPR composed of 3 angle and 10 Doppler bins. From the ideal target plot it can be seen that JDL is capable of detecting the relatively weak target correctly even though the target is buried in a region of the angle-Doppler space that is heavily dominated by both the ionospheric clutter as well as the high-power external interference. The target ∆MSMI of JDL is almost 8dB, which emphasizes the algorithm’s ability to distinguish between an ideal target and surrounding clutter.

For the realistic target scenario (Figure 2.4) JDL also provides better performance compared to the nonadaptive matched filter. However, it suffers a significant loss in performance as compared to the case of an ideal target localized to only one range bin. The spread of the target in range is accounted for by using a sufficient number of guard ranges (3 on either side of each look range) thus minimizing the contribution of ranges containing target components to the estimation of the interference covariance matrix. The peak MSMI of JDL in this case is 17.8dB and the corresponding ∆MSMI is 4dB.

The next set of figures (Figures 2.5 and 2.6) present the results from using the D^3 and hybrid algorithms to detect these injected targets. As expected due to its known weakness against correlated interference, the D^3 algorithm fails to detect both the ideal and realistic targets. On the other hand the hybrid algorithm is capable of detecting both the ideal and realistic targets. In fact, it outperforms JDL in both the realistic and ideal cases. For the ideal target (Figure 2.5) the hybrid algorithm yields an MSMI about 2.5dB larger than that of JDL, and a ∆MSMI that is 5.5dB larger. In the realistic target scenario (Figure 2.6) JDL fails to detect the target while the hybrid algorithm does so with about a 1-dB margin. To improve processing speed, we used only a quarter
2.5. PERFORMANCE OF AVAILABLE STAP ALGORITHMS IN HFSWR

Figure 2.5: Results of using the hybrid, $D^3$, JDL, and Nonadaptive MF algorithms to detect an ideal target with amplitude 55 dB inserted into the ionospheric clutter region.

Figure 2.6: Results of using the hybrid, $D^3$, JDL, and Nonadaptive MF algorithms to detect a realistic target spread over 7 range cells and with amplitude 55 dB inserted into the ionospheric clutter region.
of the number of available pulses in the data set, and we still obtained improved results for the hybrid approach. Thus the hybrid algorithm does a good job of suppressing any uncorrelated interference present in the look range after which it uses the second stage of the statistical JDL algorithm to null out any correlated interference that might be masking the target.

Again we emphasize the fact that we are performing all our calculations for a worst case target scenario, i.e., a relatively weak target buried in high-power ionospheric clutter and high-power external interference, with only a few secondary range bins from which to estimate a covariance matrix.

We next investigated the PAMF algorithm. Figures 2.7 and 2.8 plot the results of using the residual sample covariance matrix (RSC) and time average sample covariance matrix (TASC) PAMF methods, with an assumed underlying multichannel AR order of 3, to detect ideal and realistic targets injected in the ionospheric clutter region. We used an ideal target amplitude of 59dB and a realistic target amplitude of 57dB. Both the RSC-PAMF and TASC-PAMF outperform the nonadaptive processor and successfully detect the injected ideal and realistic targets. Interestingly, both these versions of the PAMF do not outperform the JDL algorithm. A possible explanation of this rather unsatisfying performance is that an AR order of 3 is used in this work, which was recognized as the optimal order for airborne radar; however, given the highly inhomogeneous structure of the clutter in the HFSWR scenario it is possible that an order of only 3 would be less than adequate to accurately model the underlying clutter statistics. Furthermore, given the limited sample support available, the JDL algorithm may benefit from the fact that the adaptive Degrees of Freedom (DoF) have been reduced significantly. It should also be noted that in our results the RSC-PAMF does a better job at detecting the injected target than the TASC-PAMF method.

In summary, the processing results indicate that JDL performs quite well in detecting both ideal and realistic targets with relatively low amplitudes (as compared to the
2.5. PERFORMANCE OF AVAILABLE STAP ALGORITHMS IN HFSWR

Figure 2.7: Results of using the RSC-PAMF, TASC-PAMF, and Nonadaptive MF algorithms to detect an ideal point target with amplitude 59 dB inserted into the ionospheric clutter region.

Figure 2.8: Results of using the RSC-PAMF, TASC-PAMF, and Nonadaptive MF algorithms to detect a realistic target spread over 7 range cells and with amplitude 63 dB inserted into the ionospheric clutter region.
ambient high-power clutter) injected in the region with dominant ionospheric clutter returns as well as high-power external interference. In both cases JDL by far outperforms the nonadaptive matched filter method and shows much better performance than the adaptive Doppler filter\(^4\). This is particularly interesting since ionospheric clutter is often considered to be too inhomogeneous for the application of statistical adaptive processing algorithms. However, as our results show, at least in the case of the Cape Race data, the data within a single data cube is sufficiently homogeneous for effective interference suppression, as long as the adaptive DoF are limited. As expected, given its inherent weakness against coherent clutter, the D\(^3\) algorithm does not perform well. The hybrid algorithm results indicate that it is the best performer among the algorithms we tested. This emphasizes the hybrid algorithm’s ability to suppress both correlated and uncorrelated interference present in the target range cell. The drawback of this hybrid approach is the computation load of the D\(^3\) stage, which at least for the time-being, makes it impractical for real time applications when \(M\) is large.

The PAMF results also show some promise. However, they cannot detect targets as weak as those detectable using the JDL or the hybrid algorithms. Although an order of 3 was shown to be optimal in the case of airborne radar, its optimality has yet to be verified for the case of ionospheric clutter. Inhomogeneous ionospheric clutter might require a larger AR order to model it accurately and is the subject of ongoing work.

We conclude that low-complexity approaches such as JDL, hybrid, and PAMF, yield useful results when there is limited secondary data, such as in the current HFSWR implementation. In our simulations we use the JDL algorithm as a baseline for comparing the performance of the new FFA algorithm. The FFA algorithm is developed in detail in the next chapter.

\(^4\)We did not include the adaptive Doppler filter simulation results so as to keep the figures readable.
Chapter 3

Fast Fully Adaptive Processing: Overview and Implementation

In the previous chapter we described several low complexity STAP methods available in the STAP literature, and assessed their performance in the HFSWR scenario. The common feature that all these methods share is that the total number of adaptive DoF are reduced to meet the constraints on the available secondary data, which inevitably leads to most of the originally available adaptive DoF being “wasted”. The JDL method, for example, adaptively processes data within a relatively small localized processing region (LPR), after performing a non-adaptive transformation of the space-time data to the angle-Doppler domain. Reducing the adaptive DoF in this manner yields corresponding reductions in required sample support and computation load, but at the same time impacts performance and reduces target discrimination.

In this chapter we will propose an alternative approach that exploits all available degrees of freedom while simultaneously reducing computational complexity and required sample support. This multistage adaptive processing scheme, which we will refer to as the Fast Fully Adaptive (FFA) approach, draws its inspiration from the butterfly structure of the Fast Fourier Transform (FFT). Essentially, the FFA approach sub-divides an $N \times M$
The key idea behind the FFA approach lies in that the outputs from each successive stage are combined adaptively to form the transform domain of the subsequent stage. This process of repartitioning the newly formed transform domain, followed by adaptively processing each resulting partition, is repeated until the original $N \times M$ data matrix is reduced to a single final statistic whose magnitude is compared against a certain threshold to determine if a target is present or absent at the range cell under test. Clearly, this approach is similar to the reduced complexity FFT algorithm. Like the FFT approach, the FFA scheme is of relatively low complexity, with the distinct advantage that the entire data matrix is adaptively processed at every stage.

In this chapter we develop the simplest FFA framework and variants of increasing complexity and performance. In the next chapter we compare the performance of the AMF (when possible), FFA, and JDL algorithms in terms of computational complexity and performance in detecting weak targets buried within a background of interference. This testing is done using two diverse interference scenarios: airborne radar and HFSWR systems. Both simulated and measured data is used. The measured HFSWR data which was obtained from a high frequency radar system operated by Defense Research and Development Canada (DRDC) at Cape Race on the Canadian East Coast, was described and analyzed in Chapter 1 and 2.

### 3.1 Regular FFA

In this section we focus on the development of the FFA approach in its most intuitive form; the regular FFA. In the previous chapter we reviewed the AMF, and described its advantages (optimal performance in the presence of ample high quality sample support),
3.1. REGULAR FFA

and disadvantages (large sample support requirements, and high complexity associated with the inversion of the interference covariance matrix). Unfortunately it is the disadvantages of this method that prevail in practice, making it virtually impossible to implement for use in real time applications. To overcome the disadvantages associated with the AMF without being forced to directly revert to a lower DoF transform domain, we will attempt to adopt a divide and conquer strategy that breaks up the $N \times M$ space-time snapshot into $(t_s \times t_t)$ smaller spatio-temporal matrices of dimensions $N' \times M'$, where $N'$ and $M'$ are chosen to satisfy $N' \ll N$ and $M' \ll M$ respectively. We then apply the AMF algorithm on each of these partitions which results in a new $t_s \times t_t$ matrix whose entries are composed of the output statistics of the corresponding AMF processes.

The weights computed by the AMF, used to process each partition, are scaled in such a manner as to maintain the target’s constant phase shift across the residual spatial and temporal dimensions. As a result the new $t_s \times t_t$ matrix of residual spatio-temporal outputs from the first stage of processing, forms a valid space-time matrix, containing the same target present in the original space-time snapshots, but with a steering vector whose inter-element and inter-pulse phase shift has been modified. Also modified by the first stage of processing is the interference, which has been suppressed in each partition by the AMF, yielding an ‘attenuated’ residual interference matrix in the forthcoming processing stage.

The resulting ‘spatio-temporal’ reduced size data-matrix is once again repartitioned (not necessarily in the same way as the original space-time snapshot) and each partition is processed by the AMF yielding the next stage of residual spatio-temporal outputs. This procedure is repeated until a final statistic is obtained, which is then compared to an appropriate threshold to decide if a target is present at the look range of interest. Note that this tree like approach of combining successively smaller problems is reminiscent of the radix-2 based FFT algorithms. Figure 3.1 illustrates this multistage formulation.

To formalize this approach, we begin with the original $N \times M$ space-time snapshot at
3.1. REGULAR FFA

Figure 3.1: A tree-like representation of the FFA method for a datacube with $M = 12$ pulses, $N = 12$ elements, spatial-partitioning-sequence=$[2,2,1,3]$, and temporal-partitioning-sequence=$[4,1,3,1]$. 
3.1. REGULAR FFA

the range of interest. We next choose an appropriate spatial and temporal partitioning scheme that partitions the $N \times M$ space-time snapshot into $t_s \times t_t$ $N' \times M'$ smaller spatio-temporal rectangles. For this partitioning scheme to be valid $t_s$ must be a factor of the current spatial dimension, $N$, and $t_t$ must be a factor of the current temporal dimension, $M$. This constraint will be touched on in the next section.

We apply the same partitioning scheme to the $N \times M$ steering matrix at the angle and Doppler of interest. We will denote the $n^{th}$ spatial and $m^{th}$ temporal partition of the space-time snapshot and steering matrix as $X_{nm}$ and $V_{nm}$ respectively where $n = 1, 2, ..., t_s$ and $m = 1, 2, ..., t_t$. An important observation is that we can write the $(n, m)^{th}$ steering matrix partition, $V_{nm}$, in terms of the first steering matrix partition, $V_{11}$, as follows,

$$V_{nm} = z_s^{(n-1)N'} z_t^{(m-1)M'} V_{11}$$  \hspace{1cm} (3.1)

where $z_s$ and $z_t$ correspond to the current spatial and temporal spacings. We begin by processing the first partition, $X_{11}$. Since the size of $X_{11}$ is $N' \times M'$, we have significantly reduced both the required sample support for covariance matrix estimation of this particular partition from $2NM$ to $2N'M'$, as well as the computation load associated with the evaluation of the inverse of this covariance matrix (reduced to $O((N'M')^3)$). We next form the optimal AMF weight vector for the first partition as follows,

$$w_{11}^{(1)} = (\hat{R}_{11}^{(1)})^{-1} v_{11}^{(1)}$$  \hspace{1cm} (3.2)

where $v_{11}$ is the vectorized form of $V_{11}$ and is of size $N'M' \times 1$. The superscript is used to indicate the current depth in the FFT tree-like structure (see Figure 3.1). At this stage it is set to 1 since we are still processing the original space-time snapshot. The next step involves the computation of the intermediate statistic associated with this first
3.1. REGULAR FFA

partition. Using Eqn.(2.1) and the corresponding partitions,

\[
y_{11}^{(1)} = \frac{(w_{11}^{(1)})^{H}x_{11}^{(1)}}{(w_{11}^{(1)})^{H}v_{11}^{(1)}}
\]
\[
= \frac{(w_{11}^{(1)})^{H}(n_{11}^{(1)})}{(w_{11}^{(1)})^{H}v_{11}^{(1)}} + \xi \frac{(w_{11}^{(1)})^{H}(v_{11}^{(1)})}{(w_{11}^{(1)})^{H}v_{11}^{(1)}}
\]
\[
= \frac{(w_{11}^{(1)})^{H}(n_{11}^{(1)})}{(w_{11}^{(1)})^{H}v_{11}^{(1)}} + \xi
\]
\[
= n_{11}^{(2)} + \xi
\]

where \( n_{11}^{(1)} \) is the colored noise component in the current partition, and where we denote the residual noise plus interference statistic as \( n_{11}^{(2)} \). \( \xi \) represents the target’s amplitude, which remains unchanged. In a similar fashion we can compute the optimal weight of the \((n, m)\)th partition as follows,

\[
w_{nm}^{(1)} = (\hat{R}_{nm}^{(1)})^{-1}v_{nm}^{(1)}.
\]

and the corresponding intermediate statistic as,

\[
y_{nm}^{(1)} = \frac{(w_{nm}^{(1)})^{H}x_{nm}^{(1)}}{(w_{nm}^{(1)})^{H}v_{11}^{(1)}}
\]
\[
= \frac{(w_{nm}^{(1)})^{H}(n_{nm}^{(1)})}{(w_{nm}^{(1)})^{H}v_{11}^{(1)}} + \xi \frac{(w_{nm}^{(1)})^{H}(v_{11}^{(1)})}{(w_{nm}^{(1)})^{H}v_{11}^{(1)}}
\]
\[
= \frac{(w_{nm}^{(1)})^{H}(n_{nm}^{(1)})}{(w_{nm}^{(1)})^{H}v_{11}^{(1)}} + \xi z_{(n-1)N'}^{(m-1)M'}
\]
\[
= n_{nm}^{(2)} + \xi z_{s}^{n-1N'}z_{t}^{(m-1)M'}
\]

where we made use of the relation in Eqn. 3.1 between the \((n, m)\)th steering matrix
partition and the first steering matrix partition. Defining $z_s^{(1)}$ and $z_t^{(1)}$ as the phase shifts corresponding to the spatial and temporal spacings of the subsequent processing stage,

$$
z_s^{(1)} = z_s^{N'}, \quad z_t^{(1)} = z_t^{M'}$$

Comparing $y_{11}^{(1)}$ and $y_{nm}^{(1)}$, it can be observed that the residual signal component of $y_{nm}^{(1)}$ is $\left[ (z_s^{(1)})^{n-1}(z_t^{(1)})^{m-1} \right]$ times the residual signal component of $y_{11}^{(1)}$. This indicates that a valid phase relationship between the residual signal components has been preserved even after undergoing the initial processing stage. As a result the $t_s \times t_t$ residual matrix can be regarded as a new space-time data matrix with a fraction of the temporal and spatial adaptive degrees of freedom (DoF) of the original space-time snapshot, and containing a target with the same amplitude but with new Doppler and angle phase shifts given by $z_t^{(1)}$ and $z_s^{(1)}$ respectively. Another important feature of this new space-time data matrix, is that its interference component is a suppressed version of the interference in the original data matrix due to the intermediate AMF filtering stage.

Choosing a new set of spatial and temporal partitions for the resulting spatio-temporal matrix at each depth, and passing each of these partitions through the AMF, will systematically lead to the evaluation of a final detection statistic which indirectly incorporates all the DoF of the original data-cube at different depths of the algorithm. However the resulting statistic is not CFAR. To overcome this problem we use the MSMI CFAR statistic as the detection statistic computed at the final stage of the algorithm where only one final spatio-temporal partition remains. This has the effect of making the final statistic CFAR. It should be noted that for the regular FFA method to yield complexity reductions, both $N$ and $M$ must have factors that are relatively small (for example both $N$ and $M$ are powers of 2 - this is similar to that required by the FFT algorithm).

A final point to note is that the FFA scheme does not lead to an equivalent model of
the AMF. As a result some performance degradation is expected in the case of sufficient sample support. However, for practical scenarios where sample support is extremely scarce, we expect the FFA scheme to yield significant improvements in performance. In the simulations section we show this to be actually true for the FFA which by far outperforms its parent algorithm, the AMF, in reduced sample scenarios.

3.2 Interleaved FFA

In this section we will develop a more general partitioning scheme for the FFA method. We will denote this new partitioning scheme as the Interleaved FFA method for reasons that will become obvious shortly.

In the previous section we introduced the FFA method which partitioned the spatial and temporal dimensions at each depth into disjoint partitions whose union resulted in the corresponding spatial or temporal dimension. However it is not essential that the partitions be disjoint, and it is possible to select a partitioning scheme in which the different spatial and/or temporal partitions overlap, thus reusing the same DoF more than once in each of the overlapping partitions.

To formalize this scheme, we begin with several definitions (refer to Figure 3.2). We define $d_N$ and $d_M$ to be the lengths of a spatial and temporal partition respectively, at a certain depth of the FFA tree. Next we define $s_N$ and $s_M$ to be the separation between two consecutive spatial and temporal partitions. Figure 3.2 might represent either the data-matrix of space-time samples at a certain depth or the steering matrix corresponding to the look angle and Doppler.

In the example shown in Figure 3.2, $d_N = 3$, $s_N = 2$, $d_M = 5$, and $s_M = 3$. If we assume the space-time matrix in Figure 3.2 represents the steering matrix at a certain depth, where $z_s$ and $z_t$ are the current spatial and temporal spacings, we can write the space-time steering matrix of the $(n, m)^{th}$ partition in terms of the first steering matrix.
3.2. INTERLEAVED FFA

$$V_{nm} = V_{11} z_{s}^{(n-1)sN} z_{t}^{(m-1)sM}$$

(3.14)

where $n \in [1, \ldots, \lfloor N/(sN + 1) \rfloor]$ and $m \in [1, \ldots, \lfloor M/(sM + 1) \rfloor]$, and where $\lfloor x \rfloor$ denotes the largest integer smaller than or equal to $x$. The remainder of the FFA method remains unchanged except that after all the spatio-temporal partitions of the current depth have been processed, we update the spatial and temporal spacings at the new depth such that,

$$z'_s = z^s_s$$

$$z'_t = z^s_t$$

(3.15)

where $z_s$ and $z_t$ correspond to the spatial and temporal spacings at the depth we just processed, and where $z'_s$ and $z'_t$ correspond to the new spatial and temporal spacings. At every new depth a different interleaved partitioning scheme can be used assuming that it is valid.
We will next address the issue of determining which combinations of \(d_N, s_N, d_M,\) and \(s_M\) are valid at a certain depth. Note that there is no requirement that \(d_N\) and \(d_M\) be factors of \(N\) and \(M\) respectively. Let \(M_i\) and \(N_i\), respectively, represent the size of the temporal and spatial dimension at depth \(i\). Clearly \(M_0 = M\) and \(N_0 = N\). Assuming we are at depth \(i\), we can write the size of the temporal dimension at the next depth, \(M_{i+1}\), in terms of the size of the spatial dimension at the current depth, \(M_i\), and the spatial partition parameters at the current depth, \(d_M(i)\) and \(s_M(i)\), as follows,

\[
M_i = d_M(i) + (M_{i+1} - 1) \times s_M(i) \quad i = 0, \ldots, t - 1
\]

\[
\Rightarrow M_{i+1} = \frac{M_i - d_M(i)}{s_M(i)} + 1
\]

where

\[
M_t = 1
\]

where \(t\) is assumed to be the final depth of the FFA tree, and \(M_0\) is the initial size of the spatial dimension of the space-time steering matrix. Similar expressions can be derived for the temporal dimension.

It is also possible to write the size of the spatial dimension at the \(i^{th}\) depth, \(M_i\), in terms of the initial spatial dimension size, \(M_0\), and the spatial partition parameters used up to the \((i - 1)^{th}\) depth, \(d_M(k)\) and \(s_M(k)\) for \(k \in [0, \ldots, i - 1]\), as follows,

\[
M_i = \left[\frac{M_0 - \sum_{k=0}^{i-1} \left( \prod_{l=0}^{k-1} s_M(l) \right) \left( d_M(k) - s_M(k) \right)}{\prod_{k=0}^{i-1} s_M(k)} \right] + 1
\]

where we define,

\[
d_M(-1) = s_M(-1) = 1
\]

A similar expression is valid for the temporal dimension. Noting that the sizes of the
spatial dimension at depth \((i + 1)\) must be less than or equal to the size of the spatial dimension at the previous depth, \(i\), we can write,

\[
1 \leq d_M(i) \leq M_i \tag{3.20}
\]

\[
1 \leq s_M(i) \leq \left\lfloor \frac{M_i - d_M(i)}{M_i - 1} \right\rfloor \tag{3.21}
\]

\[
= \left[ M_0 - \sum_{k=0}^{i-1} \left( \prod_{l=0}^{k-1} s_M(l) [d_M(k) - s_M(k)] \right) + \prod_{k=0}^{i-1} s_M(k) \frac{d_M(i)}{\prod_{k=0}^{i-1} s_M(k)} \right]
\]

\[
M_0 - \sum_{k=0}^{i-1} \left( \prod_{l=0}^{k-1} s_M(l) [d_M(k) - s_M(k)] \right) \]

\[
\text{for } i < t = \max(\text{depth}) \tag{3.22}
\]

Similar expressions can be derived for the spatial dimension. Thus upon specifying \(d_M(i)\) at a certain depth \(i\), an upper bound for \(s_M(i)\) can be selected. It can easily be verified that if we set \(s_M(i) = d_M(i)\) and \(s_N(i) = d_N(i)\) at every depth \(i\), the interleaved FFA simplifies to the regular FFA described in the previous section.

A final point worth mentioning is that it is not necessary to use all the spatial and temporal elements contained within each partition in the modified AMF solution. It is possible to only select every \(k^{th}\) spatial and every \(l^{th}\) temporal element within each spatio-temporal partition. However the same spatial and temporal sampling must be administered to all the partitions at the current depth. Clearly, though, using fewer adaptive DoF per partition, might lead to worse performance than if all the adaptive DoF were used.

This new interleaved partitioning scheme is better than the one introduced in the previous section, since the same adaptive DoF can be used in combination with different adaptive DoF to yield different intermediate statistics which better utilize the set of
available adaptive DoF at a certain depth to null the residual interference and enhance the buried target. Another advantage of this interleaved partitioning scheme is that it can be used with arbitrary values of \( N \) and \( M \) that are prime or do not admit numerous factors. Unfortunately using interleaved partitions further complicates the task of choosing an optimal spatio-temporal partition sequence, as we are no longer bound to utilizing combinations of the factors of \( N \) and \( M \), and the performance and complexity is now determined by four parameter sequences (as well as the available sample support \( K \)) instead of only two. Another drawback of using interleaved partitioning as opposed to regular partitioning, is that a larger number of partitions must be processed at every depth so as to obtain a significant improvement in performance. Unfortunately this leads to a correspondingly significant increase in running time which might lower the practical value of using this interleaved partitioning scheme.

### 3.3 Unequal Partitions

So far we have assumed that the partitions at each depth are of equal size, i.e. all the spatio-temporal ‘rectangles’ at a certain depth are of equal dimensions. Although this might seem necessary so as to be able to express all the steering matrix partitions at a given level in terms of the first steering matrix partition at this level, this requirement can be easily bypassed by using steering vectors that do not have constant phase progressions from element to element and/or pulse to pulse.

We begin by assuming unequal but disjoint spatial and temporal partitions. Starting with the original data-rectangle of dimension \( N_0 \times M_0 \) at a given look range we segment the spatial and temporal dimensions into \( N_1 \) and \( M_1 \) partitions of unequal sizes respectively. To simplify the presentation, we will require that each of the spatial and temporal partitions be disjoint. We can certainly use an interleaved partitioning scheme similar to the scheme described in the previous section to bypass the disjoint partitions.
requirement as well as utilize unequal partition sizes at any given depth of the FFA tree structure. For the time being, however, we will assume disjoint partitions so as to avoid overcomplicating an inherently simple idea. Therefore we will partition the spatial dimension into $N_1$ disjoint partitions of lengths $\left\{ n_1^{(1)}, n_2^{(1)}, \ldots, n_{N_1}^{(1)} \right\}$ where, $\sum_{i=1}^{N_1} n_i^{(1)} = N_0$ and the temporal dimension into $M_1$ disjoint partitions of lengths $\left\{ m_1^{(1)}, m_2^{(1)}, \ldots, m_{M_1}^{(1)} \right\}$ where, $\sum_{k=1}^{M_1} m_k^{(1)} = M_0$. We will denote the $(l, k)^{th}$ spatio-temporal partition of the space-time data and steering matrices at the first depth as $X_{lk}^{(1)}$ and $V_{lk}^{(1)}$ respectively, where $l = 1, 2, \ldots, N_1$ and $k = 1, 2, \ldots, M_1$. We next process each partition in a similar manner to the regular FFA method, except that in the expression pertaining to the computation of the partition’s intermediate statistic, Eqn.(3.8), we will replace the term $v_{11}^{(1)}$ in the denominator, which is an $n_1^{(1)} \times m_1^{(1)}$ vector corresponding to the first partition of the steering matrix at the current depth, by the vectorized form of the $n_1^{(1)} \times m_1^{(1)}$ steering matrix partition we would have obtained had we chosen the first partition’s spatial and temporal lengths to be $n_1^{(1)}$ and $m_1^{(1)}$ respectively. We will denote this improvised steering matrix partition as $V_{11}^{(1)}(n_1^{(1)}, m_1^{(1)})$. Thus the expression for the $(l, k)^{th}$ statistic evaluated from the $(l, k)^{th}$ spatio-temporal data partition is now given by,

$$y_{lk}^{(1)} = \frac{\left( w_{lk}^{(1)} \right)^H X_{lk}^{(1)}}{\left( w_{lk}^{(1)} \right)^H V_{11}^{(1)}(n_1^{(1)}, m_1^{(1)})} = \frac{\left( w_{lk}^{(1)} \right)^H \text{noise}_{lk}^{(1)}}{\left( w_{lk}^{(1)} \right)^H V_{11}^{(1)}(n_1^{(1)}, m_1^{(1)})} + \frac{\left( w_{lk}^{(1)} \right)^H \xi V_{lk}^{(1)}}{\left( w_{lk}^{(1)} \right)^H V_{11}^{(1)}(n_1^{(1)}, m_1^{(1)})} = \text{noise}_{ik}^{(1)} + \xi p_{ik}^{(1)} \quad (3.23)$$
where,

\[
P_{lk}^{(1)} = \frac{(w_{lk}^{(1)})^H v_{lk}^{(1)}}{(w_{lk}^{(1)})^H v_{11}^{(1)}(n_l^{(1)}, m_k^{(1)})} = \frac{(w_{lk}^{(1)})^H (z_{l}^{q_l} z_{l}^{q_m}) v_{11}^{(1)}(n_l^{(1)}, m_k^{(1)})}{(w_{lk}^{(1)})^H v_{11}^{(1)}(n_l^{(1)}, m_k^{(1)})} = \frac{z_{l}^{q_l} z_{l}^{q_m}}{q_l = \sum_{i=1}^{l-1} n_i^{(1)} \quad \text{and} \quad q_m = \sum_{i=1}^{k-1} m_i^{(1)}}
\]

is the \((l, k)^{th}\) entry of the next depth’s steering matrix, and \(y_{lk}^{(1)}\) is the \((l, k)^{th}\) entry of the next depth’s data matrix. Thus by updating the steering matrix at each depth we are not forced to rely on a preservation of a linear phase shift throughout the depth of the FFA-tree structure. In fact the steering matrix need not be characterized by any particular phase relationship amongst its entries; it suffices to know the form of the steering matrix at any given depth in order to be able to perform the FFA processing at that depth. Actually this is also true for the AMF method which does not impose a particular form that should be strictly adhered to by the steering matrix. The only strict requirement is that the steering matrix be known a priori so that it may be used in the computation of the AMF weight vector.

It is therefore possible to use unequal partition sizes at each depth, and still apply the FFA approach. A further generalization of the FFA scheme is the combination of unequal partition sizes at every depth with the use of interleaved sequences. This generalized form of the FFA could prove to be very powerful, as it no longer requires \(N\) and \(M\) to be integers with numerous factors, and it allows the possibility of DoF ‘recycling’ which improves the detection capability of the algorithm. Unfortunately, choosing a ‘good’ spatial and temporal sequence becomes more difficult when the lengths
of the spatial and temporal partitions at each depth are allowed to vary. As will be shown in the simulation results section, the performance of this FFA scheme is sensitive to the choice of partitioning scheme, and obtaining an acceptable performance requires the selection of a ‘good’ partitioning scheme. Given the numerous parameters at every depth that characterize the FFA and its variants, it is not obvious how to efficiently choose an optimal combination of parameters so as to maximize performance. In the following section we will introduce a randomized partitioning scheme that could be used to better stabilize the performance of the FFA with respect to the choice of partitioning parameters.

3.4 Randomized FFA

The primary issue with the FFA method is the absence of a systematic and efficient procedure to determine the optimal set of spatio-temporal partitions that will maximize performance of the algorithm. A computer search for the best sequence becomes impossible for the case of large values of \( N \) and \( M \) with numerous factors. This task becomes even more difficult in the interleaved-FFA scenario where the number of parameters requiring optimization has doubled. In this section we will propose a solution to this problem by introducing a new partitioning scheme, that relies on selecting the partitions at every depth in a random fashion.

As was shown in the previous section since we are not bound by any phase preservation rule we need not divide the AMF statistic by \( \left( w_{lk}^{(1)} \right)^H v_{11}^{(1)} \) to preserve a constant phase shift from element to element after each processing stage. In fact we do not even need to use rectangular partitions. We can simply choose any number of space-time elements at any random position in the current data-square and subject them to AMF processing. The resulting statistics can be grouped into a new data-vector and reprocessed again in the next stage.
Since we are free to choose as many combinations of space-time elements to process at a given depth as we want (limited by the total number of combinations, of all possible sizes, of the space-time elements of the data matrix at the current depth given by \( \sum_{k=2}^{N(d)M(d)} \binom{N(d)M(d)}{k} \gg N(d)M(d) \)), we can end up with the same, fewer, or more, number of elements at the next stage. A practical implementation might rely on randomly choosing \( H \) bundles of space-time entries at every stage, performing the modified AMF on each bundle, then regrouping the resulting statistics into a new vector. The sizes of each bundle could be chosen in such a way as to yield decreasing subsequent vector sizes until a final statistic has been obtained. One could repeat the above processing for a variety of element-bundle mappings, and averaging the final statistic.

A primary disadvantage of randomly selecting \( P \) combinations of size \( L \) each, is that since the selection is performed randomly we might not use all the DoF available at a certain depth, or even worse we might end up selecting the same combination of DoF several times (especially when \( L \) and \( P \) are large). A possibly better implementation of the randomized FFA would be to first randomly scramble the indices of all the elements of the data vector at a given depth (i.e., randomly reorder the elements within the vector), followed by sequentially partitioning the scrambled data vector into equal sized segments and then processing each segment separately with the AMF method. Of course, if the chosen partition size does not exactly divide the size of the data vector at the current depth, the last segment will be of a smaller size than all the previous segments. This is not a problem as the choice of equal length partitions was only for simplicity. Scrambling and partitioning the data vector into disjoint partitions only once at a given depth does not lead to DoF recycling as each element is used in only one segment.

To allow for reuse of the DoF at a given depth, we can repeat the above process numerous times at each depth. The advantage of doing this is that we can now reuse all
the DoF at a certain depth numerous times, thus increasing the performance of the algorithm. However the drawback is increased computational load as we are now processing a much larger number of partitions using the AMF at each depth, as well as suffering the overhead of several calls to the scrambling function whose complexity is $O(Y(d))$ where $Y(d)$ is the size of the data vector at depth $d$.

For future reference we will denote the sequence of partition sizes at each depth as the partition-sequence, and the sequence containing the number of iterations (i.e., the number of times each DoF is reused) as the iteration-sequence. We will also denote this particular randomized partitioning scheme when integrated into the FFA method as the randomized FFA scheme.

As is obvious from the above discussion this FFA method looses the FFT like structure that originally inspired its development. In fact this method has no structural pattern and becomes increasingly difficult to analyze. There are literally an infinite number of ways to process a given cube using this unstructured method, and one can no longer think in the direction of optimization. Although using this random bundle approach has the distinctive advantage of making the FFA more stable (i.e., less sensitive to the chosen partitioning scheme), it could lead to a larger computational load as we might require a large number of bundles at each stage to get an adequate improvement in the performance of the FFA.

We expect the randomized FFA to show improved performance against nonhomogeneities present in the primary range cell, as compared to the regular and interleaved partitioning schemes. We believe that the combination of the random partitions at each depth averages out any nonhomogeneities that might be present within the primary range, and thus we expect better performance of this randomized FFA approach against ionospheric clutter which is characterized by the presence of nonhomogenities due to the existence of scattered plasma layers within the ionosphere.
3.5 A General Adaptive Multistage Processing Framework

Although we have introduced the FFA scheme as a practical multistage processing approach to the AMF, it is interesting to note that the FFA actually constitutes a more general theme that can be applied to a variety of other STAP algorithms, including the JDL, D$^3$, and hybrid algorithms mentioned previously. This can be done by replacing the space-time domain on which the FFA performs its processing by any other valid transform domain (such as the angle-Doppler domain used in the JDL method), and the space-time steering matrix by the transformed steering matrix. The primary incentive in applying FFA onto a reduced DoF algorithm is to further address the issue of limited sample support. For example if we only had 20 statistically homogeneous secondary samples for covariance estimation (as is typical in practice), then using JDL with a LPR composed of 3 angle and 3 doppler bins would seem like a good choice. However if the FFA approach were incorporated into JDL for multistage processing in the transform domain (i.e., the angle-Doppler domain), we could use significantly more angle and doppler bins (say 27 angle and 27 doppler bins) while retaining a relatively small angle-doppler partition size ($3 \times 3$) to get improved performance in this low sample support scenario.

Similarly the FFA framework could be incorporated into the D$^3$ stage of the hybrid algorithm [16] which suffers from a relatively high level of complexity (especially in scenarios where $M$ is large) to significantly reduce complexity and improve running time, thus making the hybrid approach more practical for use in real time applications.

In general any adaptive algorithm that relies on the AMF at any intermediate stage, can ultimately be modified to incorporate the FFA. This generalization could be extended to methods that do not rely on the AMF solution, such as the nonstatistical D$^3$ method; however caution must be exercised in this situation as the extension involved is not quite as direct (or obvious) as for algorithms which explicitly use the AMF as a central adaptive
In interest of time, we do not investigate these generalizations any further and, in Chapter 5, suggest them as future work.

3.6 Complexity Analysis

In this section we will attempt to better characterize the advantages associated with using the FFA approach, by performing a complexity analysis of the three developed FFA partitioning schemes. The FFA method has the potential of yielding significant improvement in computation speed compared to the practically un-achievable AMF. However the real test of valor will be a comparison of the complexity of the FFA algorithm to other popular STAP algorithms that have been designed for low computation load, such as the JDL algorithm.

It is somewhat difficult to compare complexities of algorithms that have numerous defining parameters such as the number of angle and Doppler bins that dictate the complexity of the JDL method, or the spatio-temporal partition sequences choice that almost fully characterizes the complexity and performance of both the regular and interleaved FFA methods. Nonetheless, we will derive expressions that directly relate the complexity of the algorithms to their various defining parameters, in an attempt to better characterize the utility of our algorithm in practical STAP scenarios. As mentioned previously we will adopt the JDL method as a representative algorithm of the family of low complexity, low DoF, STAP methods. Both JDL and FFA schemes require limited sample support. The comparison here is in terms of computational complexity only.

Beginning with the JDL algorithm, we assume a local processing region (LPR) comprised of $\eta_a$ angle bins and $\eta_d$ Doppler bins. The algorithm running time for a single look range can be approximated as $T_{JDL} = \eta_a \eta_d (T_1) + \epsilon$ where $T_1$ is the time needed to compute 1 column entry of the JDL transformation matrix, and $\epsilon$ is the time needed to
perform inversion of the covariance matrix and is in the order of $O((\eta_a\eta_d)^3)$. However
the transformation matrix needs to be computed only once, and not for every look range.
Consequently for $r$ look ranges the running time of JDL would approximately be given
by $rO((\eta_a\eta_d)^3) + O(\eta_a\eta_d NM)$, where the term associated with $T_1$ can be safely ignored
since $T_1 \ll \epsilon$. Note that a reduction in the size of the LPR leads to a reduction in the
computation complexity associated with the JDL algorithm, but also leads to a loss in
performance.

Next we derive the complexity of the regular FFA algorithm. In essence the complexity
associated with the processing of a spatio-temporal partition of size $N' \times M'$ is that of
the AMF and is primarily defined by the complexity associated with the inversion of an
$N'M' \times N'M'$ matrix, and is therefore given by $O((N'M')^3)$. There are $t_s^{(i)} t_t^{(i)}$ partitions
at every depth, $i$, each of which must undergo AMF processing. For the regular FFA
method, $t_s^{(i)}$ must be a factor of $N(i)$ and $t_t^{(i)}$ must be a factor of $M(i)$ at every depth $i$.
Thus assuming a given spatial partitioning sequence $S_N = \{n_1, n_2, ..., n_{\text{max}(k_N,k_M)}\}$ and
a temporal partitioning sequence $S_M = \{m_1, m_2, ..., m_{\text{max}(k_N,k_M)}\}$, where $m_i$ and $n_i$ are
the spatial and temporal dimensions of the partitions at depth $i$, and $k_N$ and $k_M$ are the
largest non-unit spatial and temporal factor indices respectively. In order to make both
spatial and temporal sequences of the same size we augment the shorter sequence with
1’s until both sequences are the same length. Thus the maximum depth of the FFT-tree
is given by $\text{max}(k_N, k_M)$. As a result the complexity of the FFA per range can be written
as follows,

$$
T_{\text{reg, FFA}} = \sum_{i=1}^{\text{max}(k_N, k_M)} \left\{ \frac{NM}{\prod_{j=1}^{i} n_j m_j} O \left( (n_i m_i)^3 \right) \right\} 
$$

$$
= \sum_{i=1}^{\text{max}(k_N, k_M)} \left\{ \left( \prod_{j=i+1}^{\text{max}(k_N, k_M)} n_j m_j \right) O \left( (n_i m_i)^2 \right) \right\}
$$

(3.25)
and since this process must be repeated for all the \( r \) look ranges, the total running time complexity is given by \( rT_{reg-FFA} \).

The interleaved FFA method’s complexity is somewhat more difficult to characterize since it involves twice the number of parameters as the regular FFA method. At a certain depth \( i \), for the spatial and temporal partitions \( d_N(i) \), \( s_N(i) \), \( d_M(i) \), and \( s_M(i) \) we must process \( M'_iN'_i \) spatio-temporal partitions using the AMF method, where,

\[
M'_i = \left\lfloor \frac{M(i) - d_M(i)}{s_M(i)} + 1 \right\rfloor, \\
N'_i = \left\lfloor \frac{N(i) - d_N(i)}{s_N(i)} + 1 \right\rfloor,
\]

are the sizes of the spatial and temporal dimensions of the next depth (i.e., \( M'_i = M_{i+1} \), and \( N'_i = N_{i+1} \)). Each AMF solution requires the inversion of a \( d_N(i) \times d_N(i) \) matrix, which is \( O((d_N(i)d_M(i))^3) \). Thus the complexity of the interleaved FFA method can be shown to be,

\[
T_{int-FFA} = \sum_{i=1}^{\max(k_N, k_M)} \left\{ N'_iM'_iO((d_N(i)d_M(i))^3) \right\}
\]

(3.27)

where the expression for \( M_i \) is given in Eqn. (3.18), and that of \( N_i \) can be easily deduced from Eqn. (3.18). Note that Eqn. (3.27) reduces to Eqn. (3.25) if we set \( s_M(i) = d_M(i) \) and \( s_N(i) = d_N(i) \) at every depth \( i \).

Next we will derive an expression characterizing the complexity of the randomized FFA scheme. At a given depth \( i \), if we denote by \( L(i) \) the size of the data vector at that depth, then we are using the AMF to process \( g(i) = \lfloor L(i)/\text{partition\_sequence}(i) \rfloor \) partitions of length \( \text{partition\_sequence}(i) \), and possibly 1 partition of length \( L(i) - g(i) \times \text{partition\_sequence}(i) \), and this is repeated for \( \text{iteration\_sequence}(i) \) different re-indexed
### 3.6. COMPLEXITY ANALYSIS

Table 3.1: Complexity expressions for the AMF, JDL, Regular-FFA, Interleaved-FFA, and Randomized-FFA methods.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Complexity per look range</th>
</tr>
</thead>
<tbody>
<tr>
<td>AMF</td>
<td>$O\left((NM)^3\right)$</td>
</tr>
<tr>
<td>JDL</td>
<td>$O\left((\eta_a\eta_d)^3\right)$</td>
</tr>
<tr>
<td>Regular-FFA</td>
<td>$\sum_{i=1}^{\max(k_N,k_M)}\left{\left(\frac{\max(k_N,k_M)}{\prod_{j=i+1}^{n_jm_j}}\right) O\left((n_i m_i)^3\right)\right}$</td>
</tr>
<tr>
<td>Interleaved-FFA</td>
<td>$\sum_{i=1}^{\max(k_N,k_M)}\left{\left\lfloor\frac{N(i)-d_N(i)}{s_N(i)} + 1\right\rfloor\left\lfloor\frac{M(i)-d_M(i)}{s_M(i)} + 1\right\rfloor O\left((d_N(i)d_M(i))^3\right)\right}$</td>
</tr>
<tr>
<td>Randomized-FFA</td>
<td>$\sum_{i=1}^{\text{max_depth}}\text{iteration_seq}(i) \times \left[\frac{L(i)}{\text{partition_seq}(i)}\right] + O\left{(\text{partition_seq}(i))^3\right}$</td>
</tr>
</tbody>
</table>

versions of the data vector at the current depth. As a result the complexity associated with the randomized FFA per range can be written as,

$$T_{rand\_FFA} = \sum_{i=1}^{\text{max\_depth}}\text{iteration\_seq}(i) \times \left[\frac{L(i)}{\text{partition\_seq}(i)}\right] + O\left\{(\text{partition\_seq}(i))^3\right\} + O\left\{(a(i))^3\right\}$$

(3.28)

$$/a(i) = L(i) - g(i) \times \text{partition\_seq}(i)$$

$$g(i) = \left\lfloor\frac{L(i)}{\text{partition\_seq}(i)}\right\rfloor$$

Deriving the overall complexity for $r$ look ranges is simply achieved by multiplying $T_{rand\_FFA}$ by the number of look ranges.

As can be seen by comparing the four expressions in Table 3.1, outlining the complexities of the algorithms being evaluated, it is not possible to draw a definitive conclusion. We can manipulate the various parameters of each algorithm to favor either algorithm, in terms of complexity. However the performance is directly related to the parameters and is often characterized by an inverse relationship with the complexity, and therefore a tradeoff is inevitable.
3.6. COMPLEXITY ANALYSIS

In general, in terms of computation load, JDL has the distinct advantage in that it requires the solution of a single size-$\eta_a\eta_d$ matrix equation, while the FFA approach (in all of its variants) requires several AMF solutions. On the other hand, in terms of performance, the regular FFA algorithm uses all $NM$ adaptive DoF while the JDL is restricted to just $\eta_a\eta_d$ adaptive DoF. The interleaved and randomized FFA methods go beyond the regular FFA method and reuse the DoF at each depth in the computation of intermediate statistics, and as a result theoretically outperform the regular FFA method. Unfortunately, this improvement in performance might come at the price of a larger complexity, and again a trade-off is inevitable.

Although it is not possible to provide a global characterization of the relative complexity of the algorithms under scrutiny, we will attempt to evaluate their running time for a practical choice of sample support, and algorithm parameters. In particular we are interested in two different interference environments: airborne radar, and HFSWR. HFSWR radar is characterized by a large number of pulses. In particular for the airborne scenario we will use a datacube composed of $N = 16$ elements, and $M = 16$ pulses, while for the HFSWR case we will use a datacube composed of $N = 16$ elements, and $M = 4096$ pulses. At this stage of the analysis we are not too interested in the actual target discrimination performance of the algorithms under test, but are more interested in the running times associated with each algorithm in the scenarios being considered.

In both the airborne radar and HFSWR scenarios we will rely on a practical choice of sample support, namely 20 secondary samples per range. This rather low value is quite typical in most real time applications where due to the large spatial variation in the interference environment homogeneity of the interference statistics is lost beyond several range cells. We will focus on deriving the average running time of each algorithm per look range per look angle per look Doppler. The simulations ran on a 2.13GHz DuoCore2 processor with 2GB of RAM, running the Windows XP operating system. Beginning with the airborne radar scenario we will use the following algorithm parameters:
Table 3.2: Average Running Time of various algorithms per look range per look angle per look doppler for the airborne ($N = M = 16$) and HFSWR ($N = 16$, $M = 4096$) radar scenarios

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Airborne Radar Avg Run Time(msec)</th>
<th>HFSWR Avg Run Time(msec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>AMF</td>
<td>25.66</td>
<td>$O(2.8 \times 10^{14})$</td>
</tr>
<tr>
<td>JDL</td>
<td>0.57</td>
<td>74.12</td>
</tr>
<tr>
<td>regular-FFA</td>
<td>5.90</td>
<td>1400.90</td>
</tr>
<tr>
<td>interleaved-FFA</td>
<td>83.74</td>
<td>2034.52</td>
</tr>
<tr>
<td>randomized-FFA</td>
<td>5.41</td>
<td>15500.71</td>
</tr>
</tbody>
</table>


- JDL: 3 angle bins, 3 Doppler bins, $1/N\sqrt{2}$ angle spacing, and $1/M\sqrt{2}$ Doppler spacing.

- Interleaved FFA: $d_M = d_N = [3, 3, 3, 3, 3, 3, 3, 2]$, and $s_M = s_N = [1, 1, 1, 1, 1, 1, 1, 1]$.

- Randomized FFA: partition-sequence= $[8, 8, 4]$, and iteration-sequence= $[1, 1, 1]$

Table 3.2 shows the running times associated with each of the five methods for the airborne radar case. As can be seen from this table, the JDL has the smallest running time among the algorithms being evaluated, followed by the randomized FFA, the regular FFA, the AMF, and finally the interleaved FFA. These results indicate that for the parameters chosen the regular FFA is about 4.3 times faster than the AMF, while the randomized FFA is just about as fast as the regular FFA since we process the same number of partitions at every depth, and the partitions used by the two schemes are also of the same sizes at every depth. Once again we emphasize that this relative performance is only valid for our current choice of parameters. Unfortunately the running time of the interleaved FFA method is larger than that of the AMF for the chosen setup. However it
should be noted that the AMF method does not detect the target while the interleaved FFA method does with a margin of 12dB over the surrounding interference.

We next evaluate the algorithms in the HFSWR scenario. The parameters characterizing the algorithms in the HFSWR case are,


- JDL: 3 angle bins, 3 Doppler bins, $1/N\sqrt{2}$ angle spacing, and $1/M\sqrt{2}$ Doppler spacing.


- Randomized FFA: partition-sequence $= [8, 8, 8, 8, 4, 4]$, and iteration-sequence $= [1, 1, 1, 1, 1, 1]$

Table 3.2 shows the running times associated with each of the five methods for the HFSWR case. Given the much larger number of pulses being processed in this radar scenario, it is impossible to use the AMF scheme, whose complexity renders it completely impractical for the HFSWR setup.

As can be seen from Table 3.2 all the algorithms average running times have increased dramatically as a result of the large increase in the number of pulses. It should be noted that although we are using a LPR of the same size for the JDL method in both the HFSWR and airborne radar setups, the running time is not determined fully by the inversion operation of the error covariance in the transform domain. In fact for such large values of $M$ the overhead of transforming the space-time data matrix to the angle-Doppler domain by multiplying the data rectangle by the computed transformation matrix, can no longer be ignored, and inevitably leads to an increase in running time of the JDL algorithm in the HFSWR scenario. Among the methods being evaluated JDL has the
3.6. COMPLEXITY ANALYSIS

The shortest running time, followed by that of the regular FFA, the interleaved FFA, and the randomized FFA. Although all FFA variants yield complexities much lower than the AMF in the HFSWR scenario, they are still characterized by a much higher complexity than the JDL method for the selected set of algorithm parameters. Nonetheless we will show, in the simulation results section, that this extra complexity will be the price that must be paid in exchange for a large improvement in performance as compared to that of the JDL.

A final comment is that the primary advantage associated with using the FFA method and its variants is that it sub-divides a large problem into a series of smaller independent problems. As a result the complexity associated with the FFA can be dramatically reduced by using parallel architectures. This can be achieved by multitasking where each processor separately processes a given subset of the partitions at every depth and returns the corresponding intermediate statistics to the main processor which reassembles a new residual data matrix and allocates each processor a subset of the partitions for the subsequent processing stage. This parallelism can greatly reduce the running time of the FFA making it all the more practical in real time applications.
Chapter 4

FFA: Numerical Evaluation

In the previous chapter we introduced the FFA multistage processing approach, and developed three different partitioning schemes. Although the performance of the FFA approach (in its most general sense) can not be directly evaluated, as it constitutes a theme that can be applied to a variety of STAP methods under a variety of scenarios, we attempt to at least examine the most obvious implementation of this theme onto the AMF method, and provide a detailed assessment of its performance. In this chapter we evaluate the performance of the regular, interleaved, and randomized FFA methods in both airborne radar and HFSWR setups, using both ideal point targets localized in range and realistic targets spread over several range bins. We provide both MSMI-versus-range plots and probability of detection ($P_D$) versus SNR plots to characterize the performance of our algorithm and compare it to that of the optimal AMF and the faster JDL method. We also provide results emphasizing the capabilities of the FFA to detect targets in scenarios where insufficient sample support renders the AMF algorithm incapable of target detection.
4.1 Probability of Detection versus SNR

In this section we provide probability of detection ($P_d$) versus signal to noise ratio (SNR) plots for a chosen constant false alarm rate (CFAR) using the regular, interleaved, and randomized FFA, AMF, and JDL algorithms for the airborne scenario. Due to the absence of an accurate model for ionospheric clutter that would allow for the generation of sufficient HFSWR data for $P_d$ vs SNR plots, we generate $\Delta$MSMI versus target amplitude plots, using the measured data set, as a proxy for the evaluation of the algorithms under scrutiny in the HFSW radar scenario.

For the airborne radar case we generate our own datacubes which we create in a two step process. In the first step we generate an $324 \times 324$ covariance matrix, which includes the effects of clutter, and white noise. Next we use this template covariance matrix to generate our test datacubes. The datacube parameters used in this simulation, as well as in subsequent airborne radar simulations, are summarized in Table 4.1.

Figure 4.1, shows an angle-Doppler plot of the generated airborne data-cube at range bin 223 without any targets injected into it. The ground clutter ridge is clearly visible.

Table 4.1: Airborne Radar Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Elements $N$</td>
<td>18</td>
</tr>
<tr>
<td>Element spacing</td>
<td>$0.5\lambda$</td>
</tr>
<tr>
<td>Array transmit pattern</td>
<td>uniform</td>
</tr>
<tr>
<td>Mainbeam transmit azimuth</td>
<td>$0^\circ$</td>
</tr>
<tr>
<td>Backlobe attenuation</td>
<td>30dB</td>
</tr>
<tr>
<td>Target Doppler</td>
<td>25Hz</td>
</tr>
<tr>
<td>Pulses $M$</td>
<td>18</td>
</tr>
<tr>
<td>PRF</td>
<td>300Hz</td>
</tr>
<tr>
<td>Clutter slope $\beta$</td>
<td>1</td>
</tr>
<tr>
<td>Thermal noise power</td>
<td>0dB</td>
</tr>
<tr>
<td>Target azimuth</td>
<td>$0^\circ$</td>
</tr>
</tbody>
</table>
4.1. PROBABILITY OF DETECTION VERSUS SNR

Figure 4.1: Angle-Doppler plot of generated airborne data at range bin 223. with a slope of $\beta = 1$. We select a probability of false alarm $P_{fa} = 0.001$. The algorithm parameters used for this simulation are summarized below,

- Regular FFA: spatial-partitioning-sequence = [3, 2, 3], and temporal-partitioning-sequence = [3, 2, 3].
- JDL: 3 angle bins, 3 Doppler bins, $1/N \sqrt{2}$ angle spacing, and $1/M \sqrt{2}$ Doppler spacing.
- Interleaved FFA: $d_M = d_N = [3, 3, 3, 3, 3, 3, 3, 3, 2]$, and $s_M = s_N = [1, 1, 1, 1, 1, 1, 1, 1, 1]$.
- Randomized FFA: partition-sequence= [9, 9, 9, 9, 4], and iteration-sequence= [9, 9, 1, 1, 1].

The number of trials we chose for the Monte-Carlo simulation was $10^4$. It should be noted that for this simulation we used $10MN = 3240$ secondary ranges to estimate the covariance matrix, which is much more than the required $2NM = 648$ secondary ranges.
specified by the RMB rule. This was done to ensure that the estimated covariance for the AMF method would be very accurate. For completeness we also provide a probability of detection versus SNR plot for the MWF described in Chapter 2. The MWF is a popular multistage scheme proposed for airborne radar. Figure 4.2 plots the probability of detection versus SNR for each of the methods being evaluated.

As can be seen from the plots, the AMF method yields the best performance, followed closely by that of the full rank MWF method. It is also apparent from the results of the simulation that for the selected spatial and temporal sequences the JDL method outperforms the regular, and interleaved FFA method by approximately 7.5dB and 2dB respectively, for a probability of detection of 0.5. However the plots indicate that the randomized FFA scheme performs better than the JDL by approximately 0.5dB, and is only 1.5dB below the optimal AMF’s performance. It should be noted that the number of secondary ranges used in this simulation by far surpasses that available in practice, and therefore does not paint a realistic picture of the relative performance of the methods being evaluated. It would thus seem more sensible to perform the evaluation in the more realistic case of limited sample support.

As a result we next repeat the above simulations with a smaller number of secondary ranges for each look range. In particular, we use only 20 secondary ranges to estimate the covariance for each look range, as this number of samples is more typical in practical scenarios where the homogeneity of the ranges only holds over a relatively short range span. This reduced secondary sample support will seriously degrade the performance of the AMF method, and will allow us to evaluate the performance of the FFA method when it is used in scenarios where secondary data is scarce. It should be noted that we used the same algorithm parameters as the previous simulation. Figure 4.3 plots the probability of detection versus SNR for each of the methods in the reduced sample support (20 secondary ranges) case.

As this new figure reveals, the AMF’s performance has significantly worsened, while
4.1. PROBABILITY OF DETECTION VERSUS SNR

Figure 4.2: Probability of detection versus SNR for a $P_{FA} = 0.001$ in the airborne scenario using 3240 secondary samples.

Figure 4.3: Probability of detection versus SNR for a $P_{FA} = 0.001$ in the airborne scenario using 20 secondary samples.
4.1. PROBABILITY OF DETECTION VERSUS SNR

that of the FFA and JDL methods are affected to a lesser extent. However the performance of the JDL maintains an edge of about 11dB over the regular FFA method, and an edge of 10dB over the interleaved FFA, for a probability of detection of 0.5. The JDL also outperforms the randomized FFA by approximately 2.5dB. The AMF, in the case of reduced sample support, completely fails to detect targets with an SNR of even as high as 50dB, indicating its inherent weakness in real life scenarios where sample support is almost always scarce. The MWF also suffers the same fate as the AMF, and is unable to detect targets with SNRs of up to 50dB. Note that for this simulation we reduced the rank of the MWF from 324 to 40 = 2 × 20 in order to accommodate the low sample support environment.

The regular FFA method although it does not perform as well as the JDL method for airborne radar, shows greater immunity to reduced sample support as compared to its ‘parent’ algorithm; the AMF. This indicates the FFA’s success in significantly reducing the required sample support of STAP algorithms such as the AMF, thus making them more useful in low sample support cases that are very typical in practice.

In order to characterize the performance of the FFA method in the HFSWR scenario and compare its coherent processing gain to that of the JDL method it is important to analyze the probability of detection versus SNR plots. Another way to evaluate the performance of the FFA method, in the HFSWR scenario, in the absence of an accurate model for the ionospheric clutter is via the generation of ∆MSMI versus target amplitude plots as a satisfactory indicator of algorithm performance. We will therefore treat the combined effects of all clutter sources as colored noise. We will not attempt to generate our own HFSWR datacubes, but rely on actual measured datacubes to perform the analysis.

The reason we use target amplitude instead of SNR is that we do not have any previous knowledge about the noise variance of the measured data; however since SNR is proportional to target amplitude we use the former as a proxy for SNR. These plots
4.1. PROBABILITY OF DETECTION VERSUS SNR

Figure 4.4: $\Delta$MSMI versus target amplitude for the JDL, and FFA algorithms using $K=93$ secondary samples for covariance estimation.

would provide us with a more reliable characterization of performance than simply plotting MSMI versus range for several ideal and real injected targets. In particular these curves would provide us with information on how effectively the algorithms can detect weak targets (i.e., the ‘cutoff’ region below the threshold MSMI of the algorithm being scrutinized). It should be noted that in our simulations the magnitude of the $\Delta$MSMI is set to the difference between the peak MSMI and the second largest MSMI in dB (i.e., the ratio between the two largest peaks in the linear MSMI curve). The sign of the $\Delta$MSMI is positive if the peak MSMI and target MSMI coincide (indicating a potential detection) and negative if the peak MSMI and target MSMI are different (indicating a miss or false alarm).

We begin by presenting several of the results obtained by plotting $\Delta$MSMI versus target amplitude (both in dB). We use 93 iterations per target amplitude to obtain the curve. For each target amplitude we compute an average $\Delta$MSMI by inserting an ideal target with the specified amplitude at 93 different ranges in the region of interest, and
averaging the obtained ∆MSMIs. Ideally the averaging would have been done on different instances of the data-cube, however since we are using measured data, we have to perform the averaging on different range cells of the same data-cube. Once again the algorithm parameters used in this simulation are shown below,

- Regular FFA: spatial-partitioning-sequence = [2, 2, 2, 2], and temporal-partitioning-sequence = [4, 4, 16, 16].

- JDL: 3 angle bins, 3 Doppler bins, $1/N\sqrt{2}$ angle spacing, and $1/M\sqrt{2}$ Doppler spacing.

- Interleaved FFA: $d_M = [2, 8, 8, 2, 16], d_N = [2, 2, 2, 2, 1], s_M = [2, 8, 8, 2, 16]$, and $s_N = [2, 2, 2, 2, 1]$.

- Randomized FFA: partition-sequence = [16, 16, 16, 16], and iteration-sequence = [1, 1, 1, 1].

Figure 4.4 shows the results for the JDL and FFA methods. The JDL’s nonlinear region seems to extend well into the linear region of all three FFA methods, before itself entering into its linear region (at beyond 48dB target amplitude). The slope of both curves in their linear region is very close to 1. For the selected parameters, and among the tested FFA schemes, the Randomized-FFA shows the best performance in the linear region, followed by that of the Regular-FFA, and Interleaved-FFA respectively. It should be noted that the performance of the regular partitioning and interleaved partitioning are very close to each other for the particular choice of parameters selected, and are below that of the randomized partitioning scheme by approximately 4.4dB. The JDL shows significantly worse performance than all three FFA schemes, and is about 34dB below that of the Regular-FFA for a fixed target amplitude in the linear region of all four methods.

The large difference in ∆MSMI between the two curves emphasizes the FFA’s superior target discrimination capabilities in the HFSWR setup. It should be noted that for this
4.1. PROBABILITY OF DETECTION VERSUS SNR

Figure 4.5: \(\Delta\text{MSMI}\) versus target amplitude for the JDL, and FFA algorithms in the reduced sample support scenarios.

Simulation 93 secondary ranges were used for the covariance estimation. Although this number might seem somewhat large, it corresponds to the number of ranges dominated by ionospheric clutter in the measured HFSWR datacube. A final comment is that in this simulation we are not using any adaptive DoF recycling in the Randomized-FFA.

We next repeat the same simulation for a smaller sample support size namely, 20 secondary ranges. The algorithm parameters utilized in this simulation are given below,

- Regular FFA: spatial-partitioning-sequence = \([2, 2, 2, 2, 1, 1]\), and temporal-partitioning-sequence = \([4, 4, 4, 4, 4, 4]\).

- JDL: 3 angle bins, 3 Doppler bins, \(1/N\sqrt{2}\) angle spacing, and \(1/M\sqrt{2}\) Doppler spacing.

- Interleaved FFA: \(d_M = [4, 3, 3, 5, 4, 4, 2, 10, 8]\), \(d_N = [2, 2, 2, 2, 2, 3, 1, 1]\), \(s_M = [2, 2, 2, 2, 2, 2, 3, 1]\), and \(s_N = [2, 1, 1, 1, 1, 1, 1, 1]\).

- Randomized FFA: partition-sequence = \([10, 10, 9, 9, 9]\), and iteration-sequence = \([1, 1, 1, 1, 1]\).
The results of this simulation are shown in Figure 4.5. There are several interesting things to note from this figure; while the performance of all three FFA schemes has dropped due to the reduction in sample support, the performance of the JDL method has improved significantly. Although this may seem counterintuitive at first, we attribute the behavior of the JDL algorithm to the inhomogeneity of the ionospheric clutter across wider range spans. That is to say, the ionospheric clutter can not maintain its homogeneity over wide range spans such as 93 ranges, leading to the formation of an inaccurate estimate of the error covariance. However over shorter range spans such as 20 ranges, the ionospheric clutter does maintain a certain degree of homogeneity and as a result leads to a more accurate estimate of the error covariance for JDL, and consequently better performance. Interestingly, the FFA methods seem to behave in the opposite manner, i.e., they seems to be less sensitive to the inhomogeneity of the ionospheric clutter than JDL, and capable of better exploiting the available sample support. As this plot reveals, the regular and interleaved FFA methods enter the linear region when the target amplitude is approximately 15dB. The randomized FFA enters its linear region earlier at approximately 7dB, while the JDL begins to demonstrate linear growth at about 23dB. In the high SNR region all three FFA schemes show similar performance for the reduced sample support scenario, and outperform JDL by approximately 12.5dB.

In the next subsection we evaluate the performance of the FFA in both airborne radar and HFSWR scenarios using MSMI versus range plots for both ideal and realistic targets.

4.2 MSMI vs Range

In this section we inject ideal and real targets at specific locations in the datacubes being tested, and attempt to detect the injected targets. The main measurement of interest in these plots, is the difference between the target MSMI (assumed to be the largest) and the second largest MSMI in dB, termed $\Delta$MSMI, which we use as a sensitivity measure
4.2. MSMI VS RANGE

(i.e., the larger the $\Delta$MSMI of a certain algorithm, the greater its ability to distinguish a target from a background of interference and noise). We begin with the airborne radar case where we inject a weak ideal point target into a data-cube which we generate using the $324 \times 324$ covariance matrix used in the Monte-Carlo simulations of the previous subsection. The target of choice is a 35dB ideal target injected at range number 223, angle $0^\circ$, and Doppler frequency of 25Hz of a 7000 range data cube. It is important to note that 35dB measures the absolute amplitude of the injected target, as opposed to the SNR. For each look range we will use all the available ranges except the current look range (and target range) as secondary data to form an estimate of the covariance matrix (i.e., we use $6998 \gg 2NM = 648$ secondary ranges for covariance estimation). We thus expect the covariance estimate to be very accurate and as a result the performance of the AMF method to be almost identical to the MF performance with known covariance. The various parameters of the methods evaluated in this simulation are summarized below:

- Regular FFA: spatial-partitioning-sequence = [3, 2, 3], and temporal-partitioning-sequence = [3, 2, 3].
- JDL: 3 angle bins, 3 Doppler bins, $1/N \sqrt{2}$ angle spacing, and $1/M \sqrt{2}$ Doppler spacing.
- Interleaved FFA: $d_M = d_N = [3, 3, 3, 3, 3, 3, 3, 3, 2]$, and $s_M = s_N = [1, 1, 1, 1, 1, 1, 1, 1, 1]$.
- Randomized FFA: partition-sequence= [9, 9, 9, 9, 4], and iteration-sequence= [9, 9, 1, 1, 1].

Figure 4.6 plots the resulting MSMI versus range results for the JDL and AMF methods, while Figure 4.7 plots the results for the regular, interleaved, and randomized FFA methods. The results indicate that all five methods detect the target, however the AMF yields the best $\Delta$MSMI, and, as a result, the best performance. As is evident from
4.2. MSMI VS RANGE

Figure 4.6: MSMI vs Range plots for using the AMF, and JDL methods. 6998 secondary samples are used to estimate the covariance matrix.

Figure 4.7: MSMI vs Range plots for using the Regular FFA, Interleaved FFA, and Randomized FFA methods. 6998 secondary samples are used to estimate the covariance matrix.
4.2. MSMI VS RANGE

Figure 4.8: MSMI vs Range plots for using the AMF, JDL, and FFA methods. Only 20 secondary samples are used to estimate the covariance matrix.

the plots the JDL method does about 9dB better than the regular-FFA, and 3dB better than the interleaved FFA in the airborne radar scenario with ample sample support, for the chosen algorithm parameters. The randomized FFA method, however, rivals the performance of the JDL and is only 2dB below the performance of the optimal AMF method. Again this case is not very realistic because in practice it is seldom the case that we have such generous sample support.

A more realistic scenario is one where only a limited number of secondary ranges are available. Figures 4.8 and 4.9 plot the resulting MSMI statistic versus range results for the five methods where only 20 secondary ranges are available. We use the same algorithm parameters as in the previous example. As can be seen from the plots, the performance of the AMF heavily plummets downwards with the target completely being missed by the algorithm. The performance of the FFA and JDL also worsen, but the JDL maintains an edge of about 15dB and 7dB over the regular and interleaved FFA methods respectively, for the selected choice of partitions. The performance of the interleaved
4.2. MSMI VS RANGE

Figure 4.9: MSMI vs Range plots for using the AMF, JDL, and FFA methods. Only 20 secondary samples are used to estimate the covariance matrix.

FFA, however, is close to that of the JDL, and actually exceeds it by approximately 0.2dB for this particular instance. These results indicate that although the regular and interleaved FFA methods do much better than the AMF in the case of reduced sample support they still fall short of the JDL’s performance in the airborne radar scenario (at least for the selected choice of spatio-temporal sequences). The randomized FFA, however, performs better than the JDL in both the ample and reduced sample support scenarios.

We next conduct tests to detect injected ideal targets (identical to the one described in Section 4.1) in the measured HFSWR data cubes. We only declare a target detection to be successful if the MSMI statistic at the target range is the maximum among the look ranges being scanned for targets. In particular we inject our ideal 45dB target at range number 223 which is at the center of the ionospheric clutter region, at an angle of 35° which is within the external interference angle span, and at a Doppler of 0.18Hz which coincides with the Bragg-Line Doppler of the data-cube at hand. We also
4.2. MSMI VS RANGE

It is computationally impossible to produce MSMI versus range plots for the AMF since the inversion of the associated covariance would be on the order of $O((MN)^3) = O(2.8 \times 10^{14})$. Instead we focus on a comparison between the performance of the FFA method (and its variants) and that of JDL and nonadaptive processing. Figures 4.10, and 4.11 show the detection results for the methods being investigated.

For this simulation the algorithm parameters used are summarized below,


- JDL: 3 angle bins, 3 Doppler bins, $1/N\sqrt{2}$ angle spacing, and $1/M\sqrt{2}$ Doppler spacing.

- Interleaved FFA: $d_M = [4, 3, 3, 5, 4, 4, 2, 10, 8]$, $d_N = [2, 2, 2, 2, 2, 3, 1, 1]$, $s_M = [2, 2, 2, 2, 2, 2, 3, 1]$ and $s_N = [2, 1, 1, 1, 1, 1, 1, 1]$.

- Randomized FFA: partition-sequence = $[10, 10, 9, 9, 9]$, and iteration-sequence = $[1, 1, 1, 1, 1]$.

As can be seen from the figures, the nonadaptive algorithm fails at detecting the target, while both the FFA, using all three partitioning schemes, and JDL methods are able to successfully detect the presence of a target. However it is evident from examining the curves that the regular FFA shows about a 6dB improvement in $\Delta$MSMI over JDL. For this particular instance, all three FFA schemes seem to perform similarly. It should be noted that the FFA results, for all three partitioning schemes, in this simulation are for only one possible spatio-temporal sequence combination, which has not been optimized in any way. We therefore expect that there exist other partitioning sequences that may yield even larger improvements in performance and push the performance boundary closer to
4.2. MSMI VS RANGE

Figure 4.10: MSMI vs Range plots for using the Nonadaptive, and JDL methods to detect a 45dB ideal target in Ionospheric clutter region. All 4096 pulses are used in this simulation.

Figure 4.11: MSMI vs Range plots for using the three FFA methods to detect a 45dB ideal target in Ionospheric clutter region. All 4096 pulses are used in this simulation.
4.2. MSMI VS RANGE

that of the AMF with ample sample support. We next investigate how the FFA fares against a reduced DoF scenario where we use only a fraction of the available 4096 pulses for target detection. Figures 4.12 and 4.13 plot the MSMI versus range curves for the FFA, JDL, and nonadaptive methods using only 128 out of the 4096 available pulses (i.e., a reduction by a factor of 32). The algorithm parameters used are summarized below,

- Regular FFA: spatial-partitioning-sequence = [2, 2, 2, 2], and temporal-partitioning-sequence = [2, 4, 4, 4].
- JDL: 3 angle bins, 3 Doppler bins, \(1/N\sqrt{2}\) angle spacing, and \(1/M\sqrt{2}\) Doppler spacing.
- Interleaved FFA: \(d_M = [4, 3, 4, 4, 4]\), \(d_N = [2, 3, 2, 2, 3]\), \(s_M = [2, 2, 3, 2, 1]\), and \(s_N = [1, 2, 1, 2, 1]\).
- Randomized FFA: partition-sequence= [10, 10, 9, 3], and iteration-sequence= [1, 1, 1, 1].

Once again the nonadaptive method fails, while both the FFA and JDL methods succeed at locating the target. Note that despite the decimation in pulse, the relative performance of the methods tested remain almost unchanged with roughly a 1dB decrease in the performance of both algorithms.

We next investigate the performance of the FFA method for real target scenarios. In this simulation we inject real targets spread over several ranges, which have been extracted from actual high SINR target returns present in the data-cubes made available to us by the DRDC. This process was described earlier in Section 2.5.1. We use the same FFA and JDL parameters used in the ideal target simulation utilizing all 4096 pulses. For this simulation, we use a sample support of 30 secondary ranges. The results are shown in Figures 4.14 and 4.15, where it is evident that both the nonadaptive as well as the JDL methods fail to detect the buried real target while the regular FFA method
4.2. MSMI VS RANGE

Figure 4.12: MSMI vs Range plots for using the Nonadaptive, and JDL methods to detect a 45dB ideal target in Ionospheric clutter region. Only 128 pulses are used in this simulation.

Figure 4.13: MSMI vs Range plots for using the three FFA methods to detect a 45dB ideal target in Ionospheric clutter region. Only 128 pulses are used in this simulation.
4.3. FFA PARAMETER SELECTION

The performance of the regular FFA method is characterized by three parameters; the spatial-partitioning sequence, the temporal-partitioning sequence, and the available sample support $K$. As long as the number of secondary samples, $K$, is greater than twice
Figure 4.14: MSMI vs Range plots for using the Nonadaptive, and JDL methods to detect a 45dB real target in Ionospheric clutter region. All 4096 pulses are used in this simulation.

Figure 4.15: MSMI vs Range plots for using the three FFA methods to detect a 45dB real target in Ionospheric clutter region. All 4096 pulses are used in this simulation.
Figure 4.16: MSMI vs Range plots for using the Nonadaptive, and JDL methods to detect a 45dB real target in Ionospheric clutter region. Only 128 pulses are used in this simulation.

Figure 4.17: MSMI vs Range plots for using the three FFA methods to detect a 45dB ideal target in Ionospheric clutter region. Only 128 pulses are used in this simulation.
the maximum product pair of the spatial and temporal partitioning sequences, and the secondary data is homogeneous, then a reasonably accurate estimate of the covariance of the partition can be formed. However it is important to note that the choice of spatial and temporal sequences (as well as their ordering) can significantly impact performance of the FFA method.

To test the sensitivity of the FFA approach, we tested all possible partition sequences. The sequences are ranked in decreasing order of performance. As an example Figure 4.18 plots the $\Delta$MSMI versus the Sequence Ranking, for all possible spatio-temporal sequences with a secondary sample support of $K \leq 20$. To create this plot, we injected a 30dB ideal target into an airborne datacube (generated using the same template $18 \times 18$ covariance described in Section 4.1) at 0 degrees, and 25Hz, then used the regular FFA method with all possible spatial-temporal sequence combinations, to detect the target. Note that such a plot is only possible for scenarios where both $M$ and $N$ are relatively small, and have only a few factors. Since we only used 20 secondary samples, we only plot in Figure 4.18 the results for the sequences whose maximum sample support requirement is less than or equal to 20 samples. In this simulation we generated the space-time data matrices at each range independently, and as a result we can assume with a high degree of confidence that all the secondary ranges are statistically homogeneous. It should also be noted that we averaged the results of each spatio-temporal sequence tested over 20 independent datacube instances.

It can be clearly seen from the results of this simulation that the performance of the regular FFA method is very sensitive to the choice of partitioning sequences selected. To obtain further insight into what might be causing this rather unstable performance, we also plot in Figure 4.19 the total number of unit length spatial and temporal partitions over the entire depth of the FFA-tree versus the partitioning sequence ranking (which is determined by cross-referencing with Figure 4.18). The results seem to indicate that sequences that use spatial only or temporal only AMF processing at a larger number of
4.3. FFA PARAMETER SELECTION

Figure 4.18: $\Delta$ MSMI versus Sequence Ranking in the airborne scenario for a 30dB ideal target using the regular FFA method for a sample support size of 20 secondary samples.

Depths perform worse than sequences that exploit both the spatial and temporal dimensions simultaneously in the intermediate AMF processing at each depth. This result seems intuitive since it is well established [4] that factored spatial or temporal only adaptive processing generally performs worse than fully adaptive space-time processing (particularly when the minimum sample support requirement has been met). Nonetheless, these results only provide a crude insight into what might be causing this alleviated sensitivity to the chosen portioning sequence for the regular FFA method, and more investigations must be performed before a definite conclusion can be drawn. Note that this problem is not unique to the regular FFA method, and can also be found in other STAP methods such as the determination of an optimal set of angle and Doppler separations in the JDL method.

In section 3.4 we proposed the randomized partitioning scheme as a possible solution to the regular FFA’s increased sensitivity to the choice of spatio-temporal partitioning sequence. In the next example we further investigate the performance of the randomized
4.3. FFA PARAMETER SELECTION

Figure 4.19: Total number of unit spatial and temporal partition lengths versus Sequence Ranking in the airborne scenario for a 30dB ideal target using the regular FFA method for a sample support size of 20 secondary samples.

We used the randomized FFA with partition-sequence = [9, 10, 10, 9, 4] and iteration-sequence = [10, 10, 1, 1, 1], to detect a 35dB ideal target at 0 degrees and 25Hz, injected at range cell 50. To obtain an average performance, we iterated over 200 iterations for each of 20 independently generated datacubes. Each of the 4000 iterations used a different random permutation of the indices of the data matrix at each depth. Thus although we used a fixed partition length sequence for all the iterations, each iteration combined the available adaptive DoF at every depth differently. The obtained standard deviation in ∆MSMI statistics computed over all the iterations, is 1.9dB which is relatively small. This indicates a relative insensitivity to how the adaptive DoF are grouped into the various partitions, for fixed partition sizes at each depth from iteration to iteration. The mean MSMI versus range plot is shown in Figure 4.20. It should be noted that we used $K = 20$ secondary samples to estimate the error covariance. As can be seen from this plot the performance of the randomized FFA (on average) is significantly better than the
Figure 4.20: Average MSMI versus Range plot, obtained by averaging the randomized FFA detection results for a 35dB target over 10000 iterations. This improvement can be partially attributed to the large DoF reuse in the randomized FFA method.
Chapter 5

Conclusions and Future Work

5.1 Conclusion

In this work we were primarily concerned with the application of STAP to the HFSWR scenario. We began by describing the HFSWR system model and reviewing several of the available models for ionospheric clutter which constitutes the primary source of interference in HFSWR applications. In order to better understand the characteristics of the interference environment we were concerned with, we analyzed several measured HFSWR datasets and were able to draw important conclusions surrounding the statistical properties of the underlying interference; conclusions that would help direct our efforts in selecting appropriate methods of suppressing this interference.

We then surveyed the available adaptive processing schemes used in HFSWR, and concluded that they are characterized by a high complexity and/or do not provide an effective enough means for mitigating interference in the this radar setup. This absence of adequate adaptive interference suppression methods in the HFSWR literature coupled with the large success space-time adaptive processing schemes have enjoyed in airborne radar, motivated us to investigate the application of STAP to the HFSWR scenario. Our preliminary analysis of the measured HFSWR datacubes indicated that the limited
sample support in HFSWR, and the inhomogeneity and nonstationarity of the prevailing ionospheric clutter were the major difficulties associated with applying STAP to HFSWR. As a result, of particular interest to us were STAP schemes that were characterized by a low computational complexity, low sample support requirements, and sufficient immunity to the presence of discrete interference. Specifically, we were interested in evaluating the performance of the JDL, one dimensional D3, Hybrid, PAMF, and MWF schemes in the HFSWR scenario. We conducted several simulations on these methods, which allowed us to characterize their performance, and conclude that STAP can successfully be applied to the HFSWR setup to yield larger processing gains and better target discrimination as compared to the more traditional nonadaptive or spatial-adaptive processing schemes.

Having investigated several of the available STAP algorithms in the context of HF-SWR, the core contribution in this thesis is the introduction of a new STAP scheme, the Fast Fully Adaptive (FFA) framework, which uses a divide and conquer strategy to significantly reduce the complexity and sample support requirements of the STAP method it is being applied to, while at the same time maintain an adequate performance. We conducted a complexity analysis of the FFA algorithm and its variants and compared it to that of the low complexity JDL method. We then performed extensive simulations for both airborne radar and HFSWR scenarios that established the high target discrimination and robustness of the FFA method in a variety of situations, especially in the HFSWR scenario. We concluded with several propositions for future research that might be investigated in the context of the FFA approach.

5.2 Future Work

Despite the apparent simplicity of the multistage processing approach introduced in this dissertation, there are several challenges associated with choosing an optimal partitioning scheme. In particular for the specific case of the regular FFA method, for large values
of $M$ and $N$ there are a huge number of different combinations of spatial and temporal sequences, some of which yield impressive results while others yield less than good results. We have not investigated the problem of selecting an optimal spatio-temporal partition sequence, and this problem remains open to future research. Although the randomized partitioning scheme introduced in this paper does reduce, to a certain extent, the method’s sensitivity to the choice of spatial and temporal partitions and leads to improved performance, it has the drawback of an increased complexity and computation time.

Another point worth mentioning is that the FFA approach can be applied to a wide range of other available STAP algorithms such as the JDL, $D^3$, Hybrid, and PAMF methods. In particular if this method were integrated into the $D^3$ method and used as a first stage of the Hybrid algorithm we predict the birth of an algorithm that is capable of performing well in scenarios where the homogeneity of the secondary data is compromised by the presence of discrete interference sources or nonhomogeneities, while at the same time yielding a significant improvement in speed and reduction in complexity as compared to the currently available hybrid algorithm. Another topic that might be interesting to investigate is the replacement of the AMF as an intermediate processing stage for each partition, with the Conjugate Gradient (CG) implementation of the reduced rank Multistage Wiener Filter (MWF), which is characterized by a much lower complexity than the conventional Wiener Filter, as well as much lower sample support requirements than the former. The substitution of the AMF intermediate stage with a reduced rank MWF intermediate stage could allow the use of larger partition sizes for a fixed sample support size, $K$, thus allowing for improvements in performance of the method in typical low sample support environments.
Appendix A

A.1 Airborne Radar Interference Models

The two primary sources of external interference in the airborne radar setup are ground clutter and barrage jamming.

In [4], ground clutter is modeled as the sum of the radar echoes from concentric clutter rings centered about the vertical projection of the aircraft onto the ground. When the clutter is unambiguous in range (i.e., the unambiguous range of the radar, given by $R_u = \frac{c}{2f_R}$, is greater than the horizon range), each clutter ring corresponds to only one range and is composed of $N_c$ clutter patches each characterized by its effective radar cross section which depends on the area and ground reflectivity of the corresponding clutter patch. Note that when the radar’s unambiguous range is smaller than the horizon range, then multiple clutter rings will contribute to the overall clutter at any given range gate, and the clutter is said to be ambiguous in range. It was shown in [4] that the ground clutter’s space-time covariance matrix, under the clutter ring model, can be given by,

$$R_{GC} = \sigma^2 \sum_{i=1}^{N_c} \sum_{k=1}^{N_a} \xi_{ik} (b_{ik}b_{ik}^H) \otimes (a_{ik}a_{ik}^H)$$ (A.1)
where $N_c$ and $N_a$ are the number of unambiguous and ambiguous ranges respectively. $\sigma^2$ is the noise variance, $\xi_{ik}$ is the clutter to noise ratio (CNR) of the $(i, k)^{th}$ clutter patch, and $a_{ik}$ and $b_{ik}$ are the spatial and temporal steering vectors of the $(i, k)^{th}$ clutter patch.

The special setup of the airborne radar can be shown to lead to a fixed relationship between the Doppler($f_{ik}$) and angle($\varphi_{ik}$) of the $(i, k)^{th}$ clutter patch, in particular,

$$f_{ik} = \frac{2v_a}{\lambda} \cos(\varphi_{ik})$$  \hspace{1cm} (A.2)

where $v_a$ is the aircraft velocity, and $\lambda$ is the operating wavelength. This in turn leads to the formation of a ground clutter ridge in sin(angle)-Doppler space, with a slope of $\beta = \frac{2v_a}{\lambda f_R}$, where $f_R$ is the PRF and $d$ is the interelement spacing. When $\beta \leq 1$ the radar is said to be unambiguous in Doppler. In the unambiguous in Doppler radar case there may exist regions in the sin(angle)-Doppler space that are clutter free. On the other hand, an ambiguous in Doppler radar experiences aliasing of the observable Doppler space ([4]).

It was mentioned in [4] that the rank,$r_c$, of the ground clutter covariance can be approximated by,

$$r_c = \lfloor N + (M - 1)\beta \rfloor$$  \hspace{1cm} (A.3)

where $\lfloor \cdot \rfloor$ is the floor() operator that rounds down to the nearest integer, and where it is assumed that $\beta$ is an integer. Eqn. A.3 suggests that ground clutter rank (in the airborne scenario for integer values of $\beta$) increases linearly with the amount of Doppler ambiguity.

The other major source of interference in the airborne radar setup, is the barrage jammer. Barrage jamming refers to a jamming signal that is spatially correlated from element to element, and temporally uncorrelated from pulse to pulse. This leads to a jamming covariance given by,

$$R_J = \sigma^2 \xi_J I_M \otimes (a_J a_J^H)$$  \hspace{1cm} (A.4)
where $\sigma^2$ is the noise variance, $\xi_{ik}$ is the jammer to noise ratio, $I_M$ is the $M \times M$ identity matrix, and $a_J$ is the jammer spatial steering vector. As a result the rank of the jammer covariance is given by $M$. This result can be easily extended to the case of $J$ incoherent jammers.

A.2 Review of Low-Degree-of-Freedom STAP Algorithms

In this section we will review several popular low-DoF STAP methods developed to address the issues of sample support and computational load.

A.2.1 Joint Domain Localized Processing

Using the data model described in section 1.2.1, the spatial steering vector $a(\phi)$, defined in Eqn. (1.3) is the magnitude and phase taper received at the $N$ elements of the array due to a calibrated far field source at angle $\phi$. Due to electromagnetic reciprocity, to transmit in the direction $\phi$ the elements of the array must be excited with the conjugates of the steering vector maximize the response in the direction $\phi$. Transformation of spatial data to the angle domain, at angle $\phi$, therefore requires an inner product with the corresponding spatial steering vector. Similarly, the temporal steering $b(f)$, defined in Eqn. (1.4), vector corresponding to a Doppler frequency $f$ is the magnitude and phase taper measured at an individual element for the $M$ pulses in a CPI. An inner product with the corresponding temporal steering vector transforms time-domain data to the Doppler domain. The angle-Doppler response of the data vector $x$ at angle $\phi$ and Doppler $f$ is therefore given by

$$\tilde{x}(\phi, f) = [b(f) \odot a(\phi)]^H x,$$

(A.5)
A Local Processing Region (LPR), as shown in Fig. A.1, is formed about the signal point and interference is suppressed in this region only. The LPR covers $\eta_a$ angle bins and $\eta_d$ Doppler bins. The choice of $\eta_a$ and $\eta_d$ is independent of $N$ and $M$, i.e., the transformation to the angle-Doppler domain decouples the number of adaptive degrees of freedom from the size of the data cube. The covariance matrix corresponding to this LPR is estimated using secondary data from neighboring range cells. The adaptive weights are calculated by

$$\hat{w} = \tilde{R}^{-1}\tilde{v},$$

where the tilde $(\tilde{\cdot})$ above the scalar $x$ signifies the transform domain. Choosing a set of spatial and temporal steering vectors generates a corresponding vector of angle-Doppler domain data.

Figure A.1: An example of a $3 \times 3$ Localized processing region in the angle-Doppler domain of the JDL method.
where $\tilde{R}$ is the estimated covariance matrix corresponding to the LPR of interest. The number of adaptive unknowns is equal to $\eta_a\eta_d$. Because of the fewer unknowns involved, the covariance matrix can be accurately estimated from a limited number of secondary data samples. $\tilde{v}$ is the steering vector for the adaptive process. This vector must not be confused with either the spatial steering vector ($a$) or the temporal steering vector ($b$) of Eqns. (1.3)-(1.4). $\tilde{v}$ is the space-time vector of Eqn. (1.2) transformed to the angle-Doppler domain.

The JDL algorithm described above can be generalized as follows: Fig. A.1 shows that data from the LPR of interest is used for the adaptation process. Eqn. (A.5) indicates that the transformation from the space-time domain to the angle-Doppler domain is, in effect, an inner product with a space-time steering vector. Mathematically, the relevant transformation is therefore a pre-multiplication with a transformation matrix. For example, if the LPR covers 3 angle bins ($\phi_{-1}, \phi_0, \phi_1; \eta_a = 3$) and 3 Doppler bins ($f_{-1}, f_0, f_1; \eta_d = 3$) the transformation process is given by,

$$\tilde{x}_{LPR} = T^H x.$$  \hspace{1cm} (A.7)

Based on Eqn. (A.5), the transformation matrix $T$ is the $(NM \times \eta_a\eta_d)$ matrix

$$T = [b(f_{-1}) \otimes a(\phi_{-1}); \ b(f_0) \otimes a(\phi_{-1}); \ b(f_1) \otimes a(\phi_{-1}); \ b(f_{-1}) \otimes a(\phi_0); \ b(f_0) \otimes a(\phi_0); \ b(f_1) \otimes a(\phi_0); \ b(f_{-1}) \otimes a(\phi_1); \ b(f_0) \otimes a(\phi_1); \ b(f_1) \otimes a(\phi_1)],$$  \hspace{1cm} (A.8)

where the semi-colon ( ; ) separates the columns of the matrix. The angle-Doppler steering vector used to solve for the adaptive weights in Eqn. (A.6) is the space-time steering vector $s$ transformed to the angle-Doppler domain via the same transformation
matrix $T$, i.e.

$$\tilde{v} = T^H v.$$  \hfill (A.9)

Note that the transformation matrix defined in Eqn. (A.8) is defined for the chosen frequencies and angles without any restrictions on their values. Further, no assumption is made as to the form of the spatial and temporal steering vectors.

The adaptive weights computed in the transform domain are used to find a statistic, such as the MSMI statistic, for detection by hypothesis testing.

One primary difficulty associated with the JDL method is the selection of appropriate angle and Doppler spacings within the LPR. These spacings are important to the good performance of the JDL. Unfortunately no systematic and efficient algorithm for choosing these spacings has yet been developed, and in practice these parameters are chosen by means of an exhaustive computer search.

### A.2.2 The Direct Data Domain Algorithm

In this section we will briefly describe the $D^3$ algorithm as developed in [40]. Assuming the conventional $N$ element, $M$ pulse scenario we form the $N \times M$ data matrix $X$ in the primary range cell:

$$X = \xi V + C + N,$$  \hfill (A.10)

$$V = a(\varphi_t) \otimes b^T(f_t),$$  \hfill (A.11)

$$a(\varphi_t) = [1 \ z_s \ ... \ z_s^{(N-1)}]^T,$$  \hfill (A.12)

$$b(f_t) = [1 \ z_t \ ... \ z_t^{(M-1)}]^T,$$  \hfill (A.13)

$$z_s = e^{i2\pi(d/\lambda)\sin(\varphi_t)},$$  \hfill (A.14)

$$z_t = e^{i2\pi(f_t/f_R)},$$  \hfill (A.15)
where $\xi$ is the target amplitude, $a(\varphi_t)$ and $b(f_t)$ are the target spatial and temporal steering vectors respectively, at azimuth $\varphi_t$ and Doppler $f_t$, and $C$ and $N$ are the clutter and noise space-time matrices respectively.

Next we form the temporal residual matrix $A_t$, such that its $(i, j)^{th}$ entry is given by,

$$A_t(i, j) = X(i, j) - z_t^{-1}X(i, j + 1), \quad i = 0, \ldots, N - 1, j = 1, \ldots, M - 2 \quad (A.16)$$

If $X$ were to contain a target at look Doppler $f_t$, then the above procedure yields a residual matrix $A_t$ free from any target terms. Ideally the entries of matrix $A_t$ carry interference only terms, but due to possible beam mis-match, some negligibly small residual target information may still be present in these entries. A target with a Doppler frequency significantly off the look Doppler, $f_t$, is an example of a discrete non-homogeneity.

We will next make use of the following scalar terms in an attempt to null the discrete interference,

$$G_{wt} = \left| b_{(0:M-2)}^H w_t \right|^2 \quad (A.17)$$

$$I_{wt} = \| A_t^* w_t \|^2 \quad (A.18)$$

$$\text{SINR}_{\text{residual}} \propto \frac{G_{wt}}{I_{wt}} = \frac{w_t^H b_{(0:M-2)}^H b_{(0:M-2)}^H w_t}{w_t^H A_t^* A_t^* w_t} \quad (A.19)$$

where ( * ) denotes complex conjugation, $G_{wt}$ represents the target signal power after temporal filtering with the weight vector $w_t$, and $I_{wt}$ represents the residual interference power after temporal filtering. As a result the residual SINR can be defined as the ratio of these two terms, and our goal is to select an optimal temporal filter $w_t$ that maximizes the above SINR.

$$w_{t_{\text{opt}}} = \arg \max_{\{ w_t \}} (\text{SINR}_{\text{residual}}) \quad (A.20)$$
If \((\mathbf{A}_t^T \mathbf{A}_t^*)^{-1}\) exists, the solution to the above optimization problem can be shown to be,

\[
\mathbf{w}_{t_{\text{opt}}} = (\mathbf{A}_t^T \mathbf{A}_t^*)^{-1} \mathbf{b}_{(0:M-2)}
\]  

(A.21)

If this matrix inverse does not exist, the solution is the eigenvector corresponding to the large generalized eigenvalue of the matrix pair \(\left( \mathbf{A}_t^T \mathbf{A}_t^*, \mathbf{b}_{(0:M-2)} \mathbf{b}_{(0:M-2)}^H \right)\).

The optimal spatial weights can be obtained in a similar fashion. Define \(\mathbf{A}_s\) as the residual spatial interference matrix, whose \((i, j)^{th}\) term is given by,

\[
\mathbf{A}_s(i, j) = \mathbf{X}(j, i) - z_t^{-1} \mathbf{X}(j + 1, i)
\]  

(A.22)

Next define the target signal power, residual interference power, and SINR after spatial filtering as,

\[
\begin{align*}
G_{ws} &= |\mathbf{a}_{(0:N-2)}^H \mathbf{w}_s|^2 \\
I_{ws} &= ||\mathbf{A}_s^* \mathbf{w}_s||^2 \\
\text{SINR}_{\text{residual}} &\propto \frac{G_{ws}}{I_{ws}} = \frac{\mathbf{w}_s^H \mathbf{a}_{(0:N-2)} \mathbf{a}_{(0:N-2)}^H \mathbf{w}_s}{\mathbf{w}_s^H \mathbf{A}_s^T \mathbf{A}_s^* \mathbf{w}_s}
\end{align*}
\]  

(A.23)\hspace{1cm} (A.24)\hspace{1cm} (A.25)

We can obtain the optimal set of spatial weights by maximizing the above SINR, which yields,

\[
\mathbf{w}_{s_{\text{opt}}} = (\mathbf{A}_s^T \mathbf{A}_s^*)^{-1} \mathbf{a}_{(0:N-2)}.
\]  

(A.26)

We can then obtain the optimal spatio-temporal set of weights, \(\mathbf{w}\), as

\[
\mathbf{w} = \begin{bmatrix} \mathbf{w}_{t_{\text{opt}}} & \mathbf{0} \\ \mathbf{0} & \mathbf{w}_{s_{\text{opt}}} \end{bmatrix} \otimes \begin{bmatrix} \mathbf{0} & \mathbf{w}_{s_{\text{opt}}} \\ \mathbf{w}_{s_{\text{opt}}} & \mathbf{0} \end{bmatrix}.
\]  

(A.27)

The 0 terms represent the lost spatial and temporal degrees of freedom, respectively.
Because the algorithm does not require any secondary data, the D\(^3\) is capable of suppressing any uncorrelated interference present in the target range. However since D\(^3\) is a nonstatistical algorithm it suffers from poor performance when correlated (homogeneous) interference is present in the primary range. Also the computational load associated with computing the inverse of the \((M - 1) \times (M - 1)\) matrix \(A_t\), can be a problem when \(M\) is large.

### A.2.3 The Hybrid Algorithm

The hybrid algorithm was developed in [16] for airborne radar applications. It is a two stage algorithm whose building blocks are the D\(^3\) and JDL algorithms. The hybrid algorithm can be summarized in the following steps:

1. Form a Localized Processing Region (LPR) by choosing a set of \(\eta_a\) angle bins centered at the look angle, and \(\eta_d\) Doppler bins centered at the look Doppler.

2. For each angle-Doppler bin, \((\phi_{-i}, f_{-j})\) within the LPR form the corresponding space-time steering vector and use the D\(^3\) algorithm to obtain the adaptive weight vector \(w(\phi_{-i}, f_{-j})\) that transforms the look range from the space-time domain to the angle-Doppler domain.

3. After looping over all the LPR bins form the adaptive transformation matrix \(T_{D^3}\). For example, for 3 angle and 3 Doppler bins \(T_{D^3}\) would be given by,

\[
T_{D^3} = \begin{bmatrix}
w(\phi_{-1}, f_{-1}) & w(\phi_0, f_{-1}) & w(\phi_1, f_{-1}) \\
w(\phi_{-1}, f_0) & w(\phi_0, f_0) & w(\phi_1, f_0) \\
w(\phi_{-1}, f_1) & w(\phi_0, f_1) & w(\phi_1, f_1)
\end{bmatrix}
\]  
(A.28)

4. Transform the target range space-time vector and all the secondary ranges to the angle-Doppler domain via the transformation matrix \(T_{D^3}\).
5. Apply the JDL algorithm to the transformed domain to obtain the optimal weight vector given by,
\[ w_{hybrid} = \hat{R}^{-1}\hat{v} \] (A.29)

6. Compute the CFAR MSMI test statistic as follows,
\[ \Lambda_{\text{Hybrid}} = \left| w_{hybrid}^H x \right|^2 \]
\[ w_{hybrid}^H \hat{v} \] (A.30)

Thus the hybrid algorithm suppresses any discrete interference that might be present at the target range in its first, D^3 stage, effectively transforming the datacube to the angle-Doppler domain. It then nulls the correlated interference by computing a weight vector derived from the estimate of the error covariance matrix in the transformed domain. The main drawback of the hybrid algorithm is its large computational time, since it has to apply the D^3 algorithm for each of the \( \eta_a \eta_d \) angle-Doppler bins of the LPR.

### A.2.4 Parametric Adaptive Matched Filter (PAMF)

In this section we will present a brief description of the main concepts behind the PAMF method. Let the localized target steering vector at azimuth \( \alpha_t \) and Doppler \( f_t \) be given by the \( NM \times 1 \) space-time steering vector \( v \), and the space-time snapshot of the datacube returns at the target range be given by the \( NM \times 1 \) vector \( x \). Adopting the same setup as previously described, the adaptive matched filter solution leads to the evaluation of the following MSMI CFAR statistic,

\[ \Lambda_{\text{AMF}} = \left| v^H \tilde{R}^{-1} x \right|^2 \]
\[ v^H \tilde{R}^{-1} v \] (A.31)
The computation of the estimate $\tilde{R}$ of the error covariance matrix, and its inversion, requires large sample support as well as an impractical computational load. To overcome this the PAMF attempts to estimate $\tilde{R}^{-1}$ using a multichannel AR process of an appropriate order. Since $\tilde{R}$ is block Hermitian then there exists a unique LDL block decomposition of this matrix given by

$$\tilde{R} = ADA^H$$

(A.32)

where $A$ is a lower triangular block matrix and $D$ is a block diagonal matrix. Assuming $\tilde{R}$ to be invertible, we can take the inverse of both sides of the above equation to yield,

$$\tilde{R}^{-1} = (A^H)^{-1} D^{-1} A^{-1}$$

(A.33)

where the matrices $A^{-1}$ and $D$ can be written as,

$$A^{-1} = \begin{bmatrix}
I_N & 0_N & 0_N & \cdots & 0_N & 0_N \\
A_1^H(1) & I_N & 0_N & \cdots & 0_N & 0_N \\
A_2^H(2) & A_2^H(1) & I_N & \cdots & 0_N & 0_N \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
A_{M-2}^H(M-2) & A_{M-2}^H(M-3) & A_{M-2}^H(M-4) & \cdots & I_N & 0_N \\
A_{M-1}^H(M-1) & A_{M-1}^H(M-2) & A_{M-1}^H(M-3) & \cdots & A_{M-1}^H(1) & I_N
\end{bmatrix},$$

(A.34)

$$D = \begin{bmatrix}
D_0 & 0 & \cdots & 0 \\
0 & D_1 & \cdots & \vdots \\
\vdots & \ddots & \cdots & 0 \\
0 & \cdots & 0 & D_{M-1}
\end{bmatrix}$$

(A.35)

where the $N \times N$ submatrix $A_i^H(j)$ is the $i^{th}$ row and $(j-i)^{th}$ column block elements of matrix $A^{-1}$ ($i, j = 0, \cdots, M-1$), and is not characterized by any particular structure.
A.2. REVIEW OF LOW-DEGREE-OF-FREEDOM STAP ALGORITHMS

Note that the block diagonal matrices \( A_H^i(0) \) are equal to the \( N \times N \) identity matrix. Also the \( N \times N \) submatrices \( D_i \), \( (i = 0, \ldots, M-1) \), are the block diagonal elements of the matrix \( D \). It was shown in [17] that the set of matrices \( \{ A_H^m(0), A_H^m(m-1), \ldots, A_H^m(1) \} \) for \( m = 0, \ldots, M-1 \), are the matrix coefficients of a \( m^{th} \) order multichannel forward linear predictor for a vector process \( \{ x(m) \} \) whose covariance matrix of the \( m^{th} \) order prediction error vector is given by the \( m^{th} \) block diagonal element \( D_i \). This can be written as,

\[
\hat{x}_m(m) = -\sum_{k=1}^{m} A_H^m(k)x(m-k) \{ m = 0, \ldots, M - 1 \} \tag{A.36}
\]

\[
\epsilon_m(m) = x(m) - \hat{x}_m(m) = \sum_{k=0}^{m} A_H^m(k)x(m-k)
\]

\[
E[\epsilon_m(m)] = D_m
\]

\[
/\{ x(m) \} = \left[ \ldots x(0) \ x(1) \ldots x(m) \ldots x(M-1) \ldots \right]
\]

In the above equation the vector term \( \epsilon_m \) is the error vector of the AR(\( m \)) process. As can be seen from the above equation this error vector is the output of the \( m^{th} \) order multichannel moving average process \( \text{MA}(m) \) whose coefficients are the matrix coefficients of the corresponding multichannel AR(\( m \)) process, and whose input is the vector process \( \{ x(m) \} \).

So far our entire discussion is general and applies to any invertible, block hermitian, matrix \( R \). The reason we chose \( x \) to denote the vector process will be clear after we plug in the LDL representation of \( \tilde{R} \) into the expression for the MSI statistic.

\[
\Lambda_{MF} = \frac{|(D^{-1/2}A^{-1}v)^H(D^{-1/2}A^{-1}x)|^2}{(D^{-1/2}A^{-1}v)^H(D^{-1/2}A^{-1}v)}
\]

\[
= \frac{|(D^{-1/2}u)^H(D^{-1/2}\epsilon)|^2}{(D^{-1/2}u)^H(D^{-1/2}u)} = \frac{|s^Hr|^2}{s^Hs} \tag{A.37}
\]
\[ u = A^{-1}v \]  
\[ \varepsilon = A^{-1}x \]  
\[ s = D^{-1/2}u \]  
\[ r = D^{-1/2}\varepsilon \]

We can write the vectors \( x \) and \( \varepsilon \) as,

\[ x_{MN \times 1} = \begin{bmatrix} x^T(0) & x^T(1) & \cdots & x^T(M-1) \end{bmatrix}^T \]  
\[ \varepsilon_{MN \times 1} = \begin{bmatrix} \varepsilon^T(0) & \varepsilon^T(1) & \cdots & \varepsilon^T(M-1) \end{bmatrix}^T \]  

where \( x(m) \) is the \( N \times 1 \) spatial steering vector at the \( m^{th} \) pulse. As a result we can use Eqn. (A.37) to write \( \varepsilon(m) \) as,

\[ \varepsilon_N(1)(m) = \sum_{k=0}^{m} A_H^m(k)x(m-k) \quad \{m = 0, \cdots, M-1\} \]  

This justifies why we chose \( x \) to denote the vector process in our general formulation. Equation (A.43) describes a multichannel MA(\( m \)) process with output error vector \( \varepsilon(m) \) and input vector sequence \( x(m) \) with coefficients \( \{A_H^m(m), A_H^m(m-1), \cdots, A_H^m(1)\} \) for \( m = 0, \cdots, M-1 \). Since the coefficients of this MA(\( m \)) process are the block elements of the \( m^{th} \) row of the inverse of the triangular block matrix \( A \) resulting from the LDL decomposition of \( R \), then it follows from our previous discussion that the covariance of the \( m^{th} \) order prediction error vector, associated with the \( m^{th} \) set of matrix coefficients, has a covariance given by the \( m^{th} \) diagonal matrix block \( D_m \). Also since this is a MA process, the error vectors \( \varepsilon(m)(m = 0, \cdots, M-1) \) are uncorrelated in pairs [42].
can be written as,

\[
E[\varepsilon(m)\varepsilon^H(m)] = D_m \{m = 0, \ldots, M - 1\} \tag{A.44}
\]

\[
E[\varepsilon(m)\varepsilon^H(n)] = 0_m \{m \neq n\}
\]

\[
\Rightarrow E[\varepsilon\varepsilon^H] = D
\]

Thus we can see that the outputs of the \(M - 1\) MA processes are temporally white. This fact is useful when we attempt to compute the covariance of the \(m^{th}\) block vector, \(r(m)\), of the vector \(r\). Therefore we have,

\[
r(m) = D_m^{-1/2} \varepsilon(m)
\]

\[
= D_m^{-1/2} \sum_{k=0}^{m} A_m^H(k) x(m - k) \quad \{m = 0, \ldots, M - 1\} \tag{A.45}
\]

and,

\[
E[r(m)r^H(n)] = D_m^{-1/2} E[\varepsilon(m)\varepsilon^H(n)] D_n^{-1/2} = 0_M
\]

\[
\Rightarrow E[rr^H] = I_{MN}
\]

Thus the vectors \(r(m)\) are both temporally and spatially white. As a result the coefficient matrix \(A\) acts as a temporal whitening filter which is followed by a spatial whitening filter embodied in the matrix \(D\). Similar expressions can be obtained for the vectors \(u\) and \(s\).

In order to bypass the computation of the inverse of an estimate of \(R\) it suffices to estimate the coefficient matrix \(A^{-1}\) and residual covariance matrix \(D\), and use them in (Eqn. (A.37)) to directly compute the MSMI criterion. The first step to achieving this is to select an appropriate AR order that accurately portrays the underlying statistics.
After the order, $P$, has been selected we can rewrite the PAMF detection statistic as,

$$\Lambda_{PAMF} = \frac{\left| \sum_{m=0}^{M-P-1} s^H(m) r(m) \right|^2}{\sum_{m=0}^{M-P-1} s^H(m) s(m)} \tag{A.47}$$

$$\varepsilon_{N\times1}(m) = \sum_{k=0}^{P} A^H(k) x(m - k + P) \{m = 0, \cdots, M - P - 1\}$$

$$r(m) = D_P^{-1/2} \sum_{k=0}^{P} A^H(k) x(m - k + P) \{m = 0, \cdots, M - P - 1\}$$

where the following modifications were administered to the definitions of $\varepsilon$ and $r$ in order to obtain this new estimate of the AMF MSMI criterion:

1. retain only the vector sequences for a MA filter of order $P \leq M - 1$.

2. Let the MA filtering step be a moving window instead of a block window.

Since $A$ and $D_p$ are unknown they must be estimated via an appropriate multichannel parameter estimation method. It was shown through theory in [42] and through simulation in [17] that the best parameter estimation method for STAP is the Least Squares method (also known as the covariance method). Simulation results in [17] suggest that an order $P = 3$ ($\ll MN$) multichannel AR process is sufficient to accurately model the underlying process for airborne radar and as a result the computational cost associated with the least squares multichannel parameter estimation method (which forms the main block of the PAMF algorithm) is significantly lower than the $O((MN)^3)$ load associated with the inversion of estimate of $R$ in the AMF method. The least squares problem can
be expressed as follows,

\[
\mathbf{X}_r^H \mathbf{X}_r \mathbf{A}' = \mathbf{X}_r^H \mathbf{x}_r \quad (A.48)
\]

where the subscript \( r \) denotes the range cell of interest, and \( L \) denotes the number of pulses used in the least square solution. A necessary but insufficient condition for \( \mathbf{X}_r^H \mathbf{X}_r \) to be nonsingular is \( P \leq L/2 \). We will set \( L = \lfloor 2M/3 \rfloor \) in our simulations. Therefore for a nonsingular \( \mathbf{X}_r^H \mathbf{X}_r \) the solution to the least square problem is,

\[
\hat{\mathbf{A}}' = (\mathbf{X}_r^H \mathbf{X}_r)^{-1} \mathbf{X}_r^H \mathbf{x}_r \quad (A.49)
\]

In our simulations we use the least squares parameter estimation method as well as two different methods for the estimation of the spatial whitening block filter (i.e., the block diagonal elements of \( \mathbf{D} \)) namely the residual sample covariance matrix,

\[
\hat{\mathbf{D}}_p(m) = \frac{1}{K} \sum_{k=1}^{K} \varepsilon(n, k/H_0)\varepsilon^H(n, k/H_0) \quad \{m = 0, \ldots, M - P - 1\} \quad (A.50)
\]

and time average sample covariance matrix methods,

\[
\bar{D}_p = \frac{1}{(M - P - 1)} \sum_{m=0}^{M-P-1} \hat{D}_p(m) \quad (A.51)
\]

In the residual covariance matrix method the average is taken over range cells that do not contain target residues (thus the \( H_0 \) conditional hypothesis). In compliance with
A.2. REVIEW OF LOW-DEGREE-OF-FREEDOM STAP ALGORITHMS

Figure A.2: Block diagram of the unconstrained Wiener Filter.

Figure A.3: Block diagram of the general sidelobe canceler.

the Brennan rule it is assumed that $K \geq 2N$. The first method has the advantage of yielding a more CFAR like behavior while the second method has the advantage of larger detection probability [17].

A.2.5 Multistage Wiener Filter (MWF)

In section 2.1 we described the optimal AMF solution which is a constrained version of the more general Wiener Filter (shown in Figure A.2), where the constraint requiring that the signal in the mainbeam (i.e., in the direction of the look steering vector, $v$) be passed undistorted by the filter, is imposed. The AMF problem, can also be cast in the form of a generalized sidelobe canceler (GSC), where the observed data vector $x$ is operated on by a unitary transformation matrix $T$ defined as,

$$T = \begin{bmatrix} v^H \\ B \end{bmatrix}$$

(A.52)

to yield the desired output signal, $d_0 = v^H x$, which is the projection of the observed data vector onto the signal space (i.e., onto the direction of the steering vector $v$), and the observed noise-subspace data vector, $x_0 = Bx$, representing the interference components.
in the observed data vector. In the above expression $B$ represents an $(MN - 1) \times (MN)$ signal blocking matrix whose rows span a subspace orthogonal to the signal space. It should be emphasized that premultiplying the input vector of a Wiener filter by a nonsingular transformation matrix does not change the MMSE or output SINR of the resulting filter [19]. It can be shown that cascading this transformation matrix with the Wiener filter structure shown in Figure A.2, yields an equivalent solution to the constrained MVDR Wiener filter discussed previously. This equivalent representation of the MVDR WF, shown in Figure A.3, will prove useful in deriving the multistage Wiener Filter (MWF) [18] and [19].

The GSC weight vector is the optimal adaptive WF weight vector that provides the best estimate of the desired signal, $d_0$, from the observed noise-subspace data vector, $x_0$. The output MMSE of the GSC is that of the unconstrained WF with desired and observed signals, $d_0$ and $x_0$ respectively, and can be shown to be identical to that of the MVDR WF described previously. This GSC filter structure admits a recursive decomposition, which leads to the MWF representation shown in Figure A.4. It is demonstrated in [18] that the MMSE at each stage can be recursively represented in terms of the MMSE of a previous stage, in the backwards recursion step, as $\xi_i = \sigma_i^2 - \xi_{i+1}^{-1} |\delta_{i+1}|^2$, where $\xi_i$, $\sigma_i^2$, and $\delta_i$ represent the MMSE, desired signal variance, and magnitude of the cross-correlation between the observed and desired signal, at the $i^{th}$ stage, respectively. Table A.1 summarizes the main forward (analysis) and backward (synthesis) recurrence relations at each stage of the MWF. An efficient method to compute the blocking matrix at every stage from the corresponding signal vector at the same stage, is described in Appendix(A) of [18], and will not be elaborated on here. Another representation of the MWF is shown in Figure A.5, where the structure has been partitioned into separate analysis and synthesis filterbanks. The analysis filterbank is represented by the matrix,
Table A.1: Forward and Backward recurrence relations for the MWF.

<table>
<thead>
<tr>
<th>Forward Recursion</th>
<th>Backward Recursion</th>
</tr>
</thead>
<tbody>
<tr>
<td>$r_{x_i d_i} = E[x_id_i]$</td>
<td>$\delta_{MN-1} = r_{x_{MN-2}d_{MN-2}}$</td>
</tr>
<tr>
<td>$\delta_{i+1} = |r_{x_i d_i}|$</td>
<td>$\omega_i = \xi_i^{-1}\delta_i$</td>
</tr>
<tr>
<td>$h_{i+1} = |[r_{x_i d_i}]|$</td>
<td>$\epsilon_i = d_i - \omega_i \delta_{i+1} \epsilon_{i+1}$</td>
</tr>
<tr>
<td>$B_{i+1} = null {h_{i+1}} = D_i H_i$</td>
<td>$\xi_i = \sigma_i^{-1} - \omega_i \epsilon_{i+1} = \sigma_i^{-1} - \xi_i^{-1}</td>
</tr>
<tr>
<td>$D_i = \begin{bmatrix} I_{MN-1-i} &amp; 0 \end{bmatrix} - \begin{bmatrix} 0 &amp; I_{MN-1-i} \end{bmatrix}$</td>
<td>$H_i = diag \left( \begin{bmatrix} h_i(1) &amp; \cdots &amp; h_i(MN - i)^{-1} \end{bmatrix} \right)$</td>
</tr>
<tr>
<td>$H_i = diag \left( \begin{bmatrix} h_i(1)^{-1} &amp; \cdots &amp; h_i(MN - i)^{-1} \end{bmatrix} \right)$</td>
<td>$d_{ MN-1} = d_{ MN-1} = \epsilon_{ MN-1}$</td>
</tr>
<tr>
<td>$d_{i+1} = h_{i+1}^H x_i$</td>
<td>$\epsilon_i = d_i - \omega_i \epsilon_{i+1}$</td>
</tr>
<tr>
<td>$x_{i+1} = B_i x_i$</td>
<td>$\xi_i = \sigma_i^{-1} - \omega_i \epsilon_{i+1} = \sigma_i^{-1} - \xi_i^{-1}</td>
</tr>
<tr>
<td>$x_{MN} = d_{MN-1} = \epsilon_{MN-1}$</td>
<td>$\epsilon_i = d_i - \omega_i \epsilon_{i+1}$</td>
</tr>
</tbody>
</table>

and operates on the initial observed vector, $x_0$, to yield the desired signals at all the $(MN - 2)$ stages of the MWF.

$$L = \begin{bmatrix} h_1^H \\ h_2^H B_1 \\ \vdots \\ h_{MN-3}^H B_{MN-3} \\ \prod_{k=1}^{MN-2} B_k \end{bmatrix} \quad (A.53)$$

$$d = \begin{bmatrix} d_0 \\ d_1 \\ \vdots \\ d_{MN-1} \end{bmatrix} = \begin{bmatrix} d_0 \\ Lx_0 \end{bmatrix} \quad (A.54)$$
The synthesis filterbank, $\mathbf{U}$, is composed of $(MN - 1)$ nested scalar WF weights, where each weight admits the error signal from the previous stage as an input [18].

$$
\mathbf{U} = \begin{bmatrix}
1 & -\omega_1^* & \omega_1^* \omega_2^* & \omega_1^* \omega_2^* \omega_3^* & \cdots & (-1)^{MN-1} \prod_{k=1}^{MN-2} \omega_k^* & (-1)^MN \prod_{k=1}^{MN-1} \omega_k^* \\
0 & 1 & -\omega_3^* & \omega_3^* \omega_4^* & \cdots & (-1)^{MN-2} \prod_{k=2}^{MN-1} \omega_k^* & (-1)^{MN-1} \prod_{k=2}^{MN-1} \omega_k^* \\
0 & 0 & 0 & 1 & \cdots & 0 & 0 & \cdots & 1 \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \\
\vdots & \vdots & \vdots & & \ddots & \vdots & \vdots & & \vdots \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \\
\end{bmatrix}
$$

(A.55)
Bibliography


