ARRAY SIGNAL PROCESSING
ROBUST TO POINTING ERRORS

JIE ZHUANG

A thesis submitted in fulfilment of requirements for the degree of
Doctor of Philosophy of Imperial College London

Communications and Signal Processing Research Group
Department of Electrical and Electronic Engineering
Imperial College London
February 2011
Abstract

The objective of this thesis is to design computationally efficient DOA (direction-of-arrival) estimation algorithms and beamformers robust to pointing errors, by harnessing the antenna geometrical information and received signals. Initially, two fast root-MUSIC-type DOA estimation algorithms are developed, which can be applied in arbitrary arrays. Instead of computing all roots, the first proposed iterative algorithm calculates the wanted roots only. The second IDFT-based method obtains the DOAs by scanning a few circles in parallel and thus the rooting is avoided. Both proposed algorithms, with less computational burden, have the asymptotically similar performance to the extended root-MUSIC.

The second main contribution in this thesis is concerned with the matched direction beamformer (MDB), without using the interference subspace. The manifold vector of the desired signal is modeled as a vector lying in a known linear subspace, but the associated linear combination vector is otherwise unknown due to pointing errors. This vector can be found by computing the principal eigenvector of a certain rank-one matrix. Then a MDB is constructed which is robust to both pointing errors and overestimation of the signal subspace dimension.

Finally, an interference cancellation beamformer robust to pointing errors is considered. By means of vector space projections, much of the pointing error can be eliminated. A one-step power estimation is derived by using the theory of covariance fitting. Then an estimate-and-subtract interference canceller beamformer is proposed, in which the power inversion problem is avoided and the interferences can be cancelled completely.
Acknowledgments

My sincerest thanks go to my supervisor Prof. Athanassios Manikas, who led me into this exciting area of array signal processing, and worked with me throughout the last four years. I thank him for his guidance, assistance, patience, and interest in my work.

For generous help and technical advice acknowledgement is due to my colleagues in the communications and signal processing group: Wei (Victor) Li, Kai Luo, Zhijie Chen, Tao Wang, Georgios Efsthathopoulos, Azihananye Mengot, Yousif Kamil, Marc Willerton and Harry Commin.

For funding my three and half years at Imperial College London, I am indebted to China Scholarship Council and the UK government that jointly offered me the UK-China Scholarships for Excellence.

I wish to thank my parents who have always loved me, supported me, and encouraged me throughout all stages of my life. Also, my parents-in-law have provided me continuing support during those difficult periods.

Lastly but most importantly, I would like to express my eternal gratitude to my loving wife Yi Chen, and our sweet son Cheng-Yu Zhuang. Over the past four years they have many personal sacrifices due to my research abroad. My wife’s unbounded love, infinite patience, and commendable encouragement have always upheld me, particularly in those bad times. I also want to thank our son, for making our lives as enjoyable as my work. To them I dedicate this thesis.
Contents

Abstract 2
Acknowledgments 3
Contents 4
Publications Related to this Thesis 7
List of Figures 10
List of Tables 13
Abbreviations 14

Chapter 1. Introduction 15
  1.1 The Fundamentals of Array Signal Processing . . . . . . . . . . . 16
  1.2 Pointing Errors . . . . . . . . . . . . . . . . . . . . . . . . . . . 21
  1.3 Research Objectives and Thesis Organization . . . . . . . . . . . 22

Chapter 2. Fast root-MUSIC for Arbitrary Arrays 25
  2.1 Background of Subspace DOA Estimation . . . . . . . . . . . . . . 27
       2.1.1 Classical MUSIC . . . . . . . . . . . . . . . . . . . . . . . . 30
       2.1.2 Root-MUSIC . . . . . . . . . . . . . . . . . . . . . . . . . . . 31
  2.2 Extended root-MUSIC for Arbitrary Arrays . . . . . . . . . . . . 33
       2.2.1 Interpolated root-MUSIC . . . . . . . . . . . . . . . . . . . 33
       2.2.2 Manifold Separation Technique (MST) . . . . . . . . . . . . 34
       2.2.3 Fourier Domain (FD) root-MUSIC . . . . . . . . . . . . . . 37
  2.3 Iterative Fast root-MUSIC . . . . . . . . . . . . . . . . . . . . . . 40
  2.4 IDFT-based root-MUSIC . . . . . . . . . . . . . . . . . . . . . . . 44
2.5 Computational Complexity Analysis ........................................ 47
2.6 Simulation Studies ............................................................... 48
2.7 Summary .............................................................................. 60
2.8 Appendix 2.A Arnoldi Iteration .................................................. 62

Chapter 3. Matched Direction Beamforming based on Signal Subspace .................. 64
3.1 Received Signal Model .............................................................. 65
3.2 Previous Work ....................................................................... 69
  3.2.1 MDB with the Knowledge of Interference Subspace ................. 69
  3.2.2 Multirank MVDR Beamformer ........................................... 70
3.3 Proposed Matched Direction Beamformer .................................... 73
  3.3.1 Estimate using Signal Subspace ......................................... 73
  3.3.2 Some Discussions ............................................................. 77
3.4 Simulation Studies .................................................................. 78
3.5 Summary .............................................................................. 85
3.6 Appendix 3.A The first two derivatives of the manifold vector .......... 87

Chapter 4. Interference Cancellation Beamforming Robust to Pointing Errors .............. 88
4.1 Introduction and Literature Review .......................................... 89
  4.1.1 Wiener-Hopf and Capon Beamformers ................................. 91
  4.1.2 Modified Wiener-Hopf ...................................................... 93
  4.1.3 Robust Techniques for Wiener-Hopf processors ................. 97
  4.1.4 Blocking Matrix Approaches ......................................... 100
  4.1.5 Problem Statement .......................................................... 104
4.2 Proposed Interference Cancellation Beamformer ............................ 105
  4.2.1 Estimate Using Vector Subspace Projections (VSP) ......... 106
  4.2.2 Power Estimator ............................................................ 108
  4.2.3 Beamformer Construction ............................................. 111
  4.2.4 Proposed Algorithm ....................................................... 113
4.3 Performance Analysis in the Presence of Pointing Errors ............... 113
4.4 Simulation Studies .................................................................. 116
4.5 Summary .............................................................................. 121
4.6 Appendix 4.A Revisit the power estimation ................................. 123
Publications Related to this Thesis


Mathematical Notations

$A, a$  Scalar

$A, a$  Vector

$A, a$  Matrix

$(-)^T$  Transpose

$(-)^H$  Conjugate transpose

$(-)^*$  Complex conjugate

$|a|$  Absolute value of scalar

$\|a\|$  Euclidian norm of vector

$N$  Number of sensors

$\mathbb{C}$  Field of complex numbers

$\mathbb{R}$  Field of real numbers

$\theta$  Azimuth angle

$\phi$  Elevation angle

$\mathbb{I}_N$  $N \times N$ Identity matrix

$\mathbb{O}$  Matrix of zeros
\( \mathbb{P} \)  
Projection operator

\( \mathbb{P}^\perp \)  
Complement projection operator

\( \mathcal{L}[\mathbb{A}] \)  
Linear subspace spanned by the columns of \( \mathbb{A} \)

\( \mathcal{S} \)  
Manifold vector

\( \sigma_d^2 \)  
Powers of the desired signal

\( \sigma_i^2 \)  
Powers of the \( i^{th} \) interference

\( \sigma_n^2 \)  
Powers of the noise

\( \mathcal{E}\{\cdot\} \)  
Expectation operator

\( eig_i(\mathbb{A}) \)  
The \( i^{th} \) largest eigenvalue of \( \mathbb{A} \)

\( eig_{\text{max}}(\mathbb{A}) \)  
The maximum eigenvalue of \( \mathbb{A} \)

\( \dim\{\cdot\} \)  
The dimension

\( \mathcal{P}\{\mathbb{A}\} \)  
The principal eigenvector of \( \mathbb{A} \)

\( \text{tr}\{\mathbb{A}\} \)  
Trace of \( \mathbb{A} \)

\( [\mathbb{A}]_{m,n} \)  
The \( (m,n)^{th} \) entry of \( \mathbb{A} \)

\( [\mathbb{A}]_n \)  
The \( n^{th} \) entry of \( \mathbb{A} \)
List of Figures

1.1 Plane wave propagation model ........................................ 18
1.2 General block diagram of an array system .......................... 20
1.3 Effects of pointing errors. The true DOA of the desired signal is 90° 22

2.1 Effect of error on the root-MUSIC .................................... 33
2.2 The angular spectra of MUSIC and LS-root-MUSIC using theoretical $\mathbb{R}_{xx}$. The true DOAs are $[90°, 95°]$. The input SNR=20dB. 49
2.3 Roots for the extended root-MUSIC and the proposed iterative fast root-MUSIC. The theoretical $\mathbb{R}_{xx}$ is used. The true DOAs are $[90°, 95°]$. The input SNR=20dB. 51
2.4 The 2-D spectrum of the proposed IDFT-based method (and contour diagram), where $\rho \in [0.5, 1]$ and $\Delta \rho = 0.01$. The theoretical $\mathbb{R}_{xx}$ is used. The true DOAs are $[90°, 95°]$. The input SNR=20dB. 52
2.5 The angular spectra of MUSIC and LS-root-MUSIC. $\mathbb{R}_{xx}$ is formed by using 100 snapshots of one realization. The true DOAs are $[90°, 95°]$. The input SNR=20dB. 54
2.6 Roots for the extended root-MUSIC and the proposed iterative fast root-MUSIC. $\mathbb{R}_{xx}$ is formed by using 100 snapshots of one realization. The true DOAs are $[90°, 95°]$. The input SNR=20dB. 55
2.7 The 2-D spectrum of the proposed IDFT-based method (and contour diagram), where $\rho \in [0.5, 1]$ and $\Delta \rho = 0.01$. $\mathbb{R}_{xx}$ is formed by using 100 snapshots of one realization. The true DOAs are $[90°, 95°]$. The input SNR=20dB. 56
2.8 DOA estimation RMSEs versus \((\theta_2 - \theta_1)\) with the snapshot number
\(= 100\), input SNR=20dB, \(\theta_1 = 90^\circ\), \(\rho \in [0.9, 1]\) and \(\Delta \rho = 0.01\). 58
2.9 DOA estimation RMSEs versus the snapshot number with input
SNR=20dB, DOAs=[90°, 95°], \(\rho \in [0.9, 1]\) and \(\Delta \rho = 0.01\). 59
2.10 DOA estimation RMSEs versus the input SNR with
DOAs=[90°, 95°], the snapshot number \(=100\), \(\rho \in [0.9, 1]\)
and \(\Delta \rho = 0.01\). 60
3.1 The GSC structure of the multirank MVDR beamformer 71
3.2 Array output SNIR versus the azimuth of the 2nd interference. The
1st interference remains at 100°. The actual DOA of the desired
signal is 90° while the presumed DOA is 87°. 80
3.3 Array output SNIR versus the azimuth of the 2nd interference.
The 1st interference remains at 97°. The actual DOA of the desired
signal is 90° while the presumed DOA is 87°. 82
3.4 Array output SNIR versus the input SNR. The two interference
DOAs are at \([80^\circ, 100^\circ]\). The actual DOA of the desired signal is
90° while the presumed DOA is 87°. All signals are equally powered. 82
3.5 Array output SNIR versus the nominal signal subspace dimension.
The true dimension is 3. The actual DOA of the desired signal is
90° while the presumed DOA is 87°. The DOAs of interferences
are \([80^\circ, 100^\circ]\). 84
3.6 Array output SNIR versus the snapshot number. The actual DOA
of the desired signal is 90° while the presumed DOA is 87°. The
DOAs of interferences are \([80^\circ, 100^\circ]\). 84
3.7 RMSE of the desired manifold versus the snapshot number. The
interference signals are from \([80^\circ, 100^\circ]\). The actual DOA of the
desired signal is 90° while the presumed DOA is 87°. 85
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1</td>
<td>Decomposition of the space spanned by the columns of the data covariance matrix when the desired signal is present</td>
</tr>
<tr>
<td>4.2</td>
<td>Decomposition of the space spanned by the columns of the data covariance matrix when the desired signal is absent</td>
</tr>
<tr>
<td>4.3</td>
<td>The structure of the beamformer using blocking matrix</td>
</tr>
<tr>
<td>4.4</td>
<td>An example of spatial blocking filter with $q = 1$</td>
</tr>
<tr>
<td>4.5</td>
<td>Illustration of the convergence of the VSP method. The point-A represents the nominal manifold. The point-B represents the true manifold.</td>
</tr>
<tr>
<td>4.6</td>
<td>Decomposition of the space by the columns of $\hat{R}_{t+n}$</td>
</tr>
<tr>
<td>4.7</td>
<td>Array pattern with two interferences close together (60° and 62°). The DOA of the desired signal is 90°.</td>
</tr>
<tr>
<td>4.8</td>
<td>Effects of pointing errors with noise level at -10dB. The true direction of the desired signal maintained at 90°.</td>
</tr>
<tr>
<td>4.9</td>
<td>Effects of pointing errors. The true direction of the desired signal maintained at 90°.</td>
</tr>
<tr>
<td>4.10</td>
<td>The effect of finite snapshots. The true direction of the desired signal maintained at 90°. The pointing angle is 88°.</td>
</tr>
</tbody>
</table>
# List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>The roots of the extended root-MUSIC and the proposed iterative fast root-MUSIC when the theoretical covariance matrix $\mathbb{R}_{xx}$ is used.</td>
</tr>
<tr>
<td>2.2</td>
<td>$(\rho, \theta)$ pairs of the peaks of LS-root-MUSIC and the proposed IDFT-based method when the theoretical covariance matrix $\mathbb{R}_{xx}$ is used.</td>
</tr>
<tr>
<td>2.3</td>
<td>The roots of the extended root-MUSIC and the proposed iterative fast root-MUSIC when the covariance matrix $\mathbb{R}_{xx}$ is formed by using 100 snapshots of one realization.</td>
</tr>
<tr>
<td>2.4</td>
<td>$(\rho, \theta)$ pairs of the peaks of LS-root-MUSIC and the proposed IDFT-based method when the covariance matrix $\mathbb{R}_{xx}$ is formed by using 100 snapshots of one realization.</td>
</tr>
<tr>
<td>4.1</td>
<td>Estimated power of the desired signal</td>
</tr>
<tr>
<td>4.2</td>
<td>SNIR$_{\text{out}}$ (unit: dB) with noise level at -10dB</td>
</tr>
</tbody>
</table>
Abbreviations

AWGN  additive white Gaussian noise
DL    diagonal loading
DOA   direction-of-arrival
FD    Fourier domain
GSC   generalized sidelobe canceller
IDFT  inverse discrete Fourier transform
IS    interference subspace
MDB   matched direction beamformer
ML    maximum likelihood
MST   manifold separation technique
MUSIC multiple Signal Classification
MVDR  minimum-variance-distionless-response
RCB   robust Capon beamformer
SNIR  signal-to-noise-plus-interference ratio
SNR   signal-to-noise ratio
SSP   signal-subspace projection
ULA   uniform linear arrays
VSP   vector space projections
Chapter 1

Introduction

ARRAY signal processing has long been of great research interest given its important role in a great variety of military and commercial applications, such as in radar, wireless communications, sonar, acoustics, astronomy, seismology, biomedicine, etc. The array systems that human is the most familiar with may be human eyes and ears, which can be viewed as two-sensor optical and acoustic arrays. Human eyes can detect the electromagnetic radiation in a band between 450THz to 750THz roughly [1]. This propagating radiation is then processed by human brain so that the outside world can be recognized by human. Similarly human ears collect the audio waves with range between 20Hz and 20,000Hz that are utilized to extract information of interest, e.g., the direction from which sound waves originate.

In order to extend Humankind’s senses, a number of sensors (transducing elements, antennas, receivers, etc) are deployed in a 3-dimension Cartesian space to measure propagating waveforms. More precisely, if these sensors share a common reference point, an array is formed. Generally the environment where the arrays are operating consists of multiple emitting sources plus noise simul-
The Fundamentals of Array Signal Processing

1.1 The Fundamentals of Array Signal Processing

1. Parameter estimation problem — where the number and direction-of-arrival (DOA) of the incident signals are of great interest. Essentially, the number and DOAs are, respectively, the spatial analogues of model order selection and frequency estimation in time-series analysis.

2. Interference cancellation — the reproduction of the desired signal from a particular direction and the cancellation of the unwanted interferences (or jammers), from all other directions, as much as possible.

3. Tracking — where the target sources are moving in space. The tracking algorithms aim at determining the source location and motion over a long period. Generally, Kalman filter and its extended versions are useful to deal with tracking problems.

In this thesis, the emphasis are placed on the DOA estimation and interference cancellation.

1.1 The Fundamentals of Array Signal Processing

The “core” of any array application is the structure of the array, which is completely characterized by the array manifold \([2]\). The array manifold is defined
as the locus of the all response vectors, by means of which one or more real directional parameters are mapped to a \((N \times 1)\) complex vector (where \(N\) is the number of sensors). As shown in Figure 1.1, a signal source is located at the far field of the array and the plane wave can be considered to model the wave propagation. The directions of the propagating wave are termed as the azimuth angle \(\theta \in [0, 2\pi)\), measured anticlockwise from the x-axis, and the elevation angle \(\phi \in [0, \pi)\), measured anticlockwise from the x-y plane. Then the received signal at the zero-phase reference point (or the original point) can be expressed as

\[
\text{signal at the reference point: } m(t) \exp(j2\pi F_c t) \quad (1.1)
\]

where \(m(t)\) denotes the complex envelope of the signal and \(\exp(j2\pi F_c t)\) stands for the carrier. Note that all the sensors are assumed to be isotropic, meaning that the sensor gain is identical at all directions. The propagation delay between the reference point and the \(n^{th}\) sensor is given by \([2]\)

\[
\tau_n = \frac{r_n^T u(\theta, \phi)}{c} \quad (1.2)
\]

where \(r_n = [x_n, y_n, z_n]^T \in \mathbb{R}^{3×1}\) represents the Cartesian coordinates associated with the \(n^{th}\) sensor, \(c\) is the wave propagation speed, say light speed, and \(u(\theta, \phi)\) is a unit-norm vector pointing towards the propagation direction \((\theta, \phi)\), i.e.,

\[
u(\theta, \phi) = [\cos \theta \cos \phi, \sin \theta \cos \phi, \sin \phi]^T \quad (1.3)
\]

Based on the above, the signal received at the \(n^{th}\) sensor can be expressed as

\[
\text{signal at the } n^{th} \text{ sensor: } m(t - \tau_n) \exp(j2\pi F_c (t - \tau_n)) \quad (1.4)
\]

which is a delayed version of the signal collected at the reference point.
The assumption that the incident signals are narrowband is used throughout this thesis. This implies that the envelope, \( m(t) \), does not change significantly as it traverses the array, and hence the approximation \( m(t - \tau_n) \approx m(t) \) holds \([2–5]\). Thus the difference between the \( n^{th} \) sensor and the reference point is nothing but a phase delay. Therefore, these different phase components at all sensors can be termed by a vector, written analytically as

\[
S(\theta, \phi) = \exp\left(-\frac{j2\pi F_c}{c} [r_1, r_2, \ldots, r_N]^T u(\theta, \phi)\right)
\]

\[
= \exp\left(-\frac{j2\pi F_c}{c} [r_x, r_y, r_z] u(\theta, \phi)\right) 
\]  

(1.5)
where $r_x$, $r_y$, and $r_z$ are $N \times 1$ vectors with elements the $x$, $y$ and $z$ coordinates of the $N$ sensors. Thus $[r_1, r_2, \ldots, r_N] = [r_x, r_y, r_z]^T \in \mathbb{R}^{3 \times N}$ contains the geometry information of all the array sensors. The $N \times 1$ vector $S(\theta, \phi)$ is the so called array manifold vector.

In many practical applications, the propagating waves arrive the array from approximately the same elevation so, without loss of generality, in this thesis $\phi = 0^\circ$ is assumed and the parameter $\phi$ is dropped in $S(\theta, \phi)$. If the unit of sensor coordinates is half wavelength, the array manifold can be rewritten as

$$S(\theta) = \exp(-j\pi [r_x, r_y, r_z] u(\theta)) = \exp(-j\pi (r_x \cos \theta + r_y \sin \theta)) \quad (1.6)$$

In the case of linear array with unit of half wavelength, the $y$ coordinate is $r_y = 0$, and thus the array manifold can be further simplified to

$$S(\theta) = \exp(-j\pi r_x \cos \theta) \quad (1.7)$$

Assume $M$ signal sources impinge the array simultaneously. The baseband (down-converted) signal-vector $\underline{x}(t)$ received at the array sensors can be expressed as [6] [7]

$$\underline{x}(t) = \sum_{i=1}^{M} S(\theta_i)m_i(t) + \underline{n}(t) \quad (1.8)$$

where $\underline{n}(t)$ denotes the noise component. Each $(N \times 1)$ vector $\underline{x}(t)$ at a fixed time $t$ is also called the snapshot at time $t$.

Figure 1.2 depicts a typical array system, where signals from each sensor are multiplied by a complex weight vector $\underline{w} \in \mathbb{C}^N$ and summed to form the array
output $y(t)$, i.e.,

$$y(t) = w^H x(t)$$

$$= \sum_{i=1}^{M} w^H S(\theta_i)m_i(t) + w^H n(t)$$

(1.9)

where the first term of the right hand side denotes the operation of the beamformer on the manifold vectors, and the second term the operation on the noise.

A number of strategies have been developed to form the weight vector so as to fulfil the required criteria. For instance, maximization of the array output signal-to-noise-plus-interference ratio (SNIR) may be the most popular measurement for the weight design. The general idea is to let the desired signal, at a particular direction (which is a known priori or can be estimated), pass through unchanged, and the unwanted interference signals be suppressed as much as possible, such that the output SNIR is as large as possible. Consequently, the associated beam pattern presents a peak at the direction of the desired signal and nulls at the directions of interferences. This process is also known as beamforming.
For processing wideband signals, a tapped-delay-line (TDL) structure is used in the beamformer normally [7].

1.2 Pointing Errors

In practical applications, the actual DOA associated with the desired signal often differs from the presumed (or nominal) one used by the array processor, which leads to

\[ \mathcal{S}(\theta_d) \neq \mathcal{S}(\theta_0) \quad \text{with} \quad \theta_d \neq \theta_0 \]  

(1.10)

where \( \theta_d \) and \( \theta_0 \), respectively, denote the actual and nominal DOA of the desired signal. This problem is known as the “pointing error” problem. It has been shown in [8] that even a slight mismatch between \( \theta_d \) and \( \theta_0 \) may result in substantial performance degradation of the conventional adaptive array processor. Let us look at an example. Assume a uniform linear array (ULA) with \( N = 10 \) sensors and half-wavelength sensor spacing operates in the presence of three source signals where one is the desired signal with DOA \( \theta_d = 90^\circ \) and two are interferences with DOAs \([60^\circ, 70^\circ]\). All three signals have powers equal to one while the noise power is 0.1. The output SNIR of the conventional adaptive array beamformer against the pointing angle (or the presumed DOA) is plotted in Figure 1.3. It can be seen that in this example the conventional adaptive beamformer fails even when the pointing error is only \( 1^\circ \).
1.3 Research Objectives and Thesis Organization

The primary objective of this thesis is to propose new DOA estimation algorithms and beamformers, robust to pointing errors. The methods developed aim at the applications at the receiver end. Throughout this thesis, it is assumed that the incident signals are located at the far field of the array, and all signals are uncorrelated to each other.

The rest of this thesis contains three technical chapters (Chapter 2, 3 and 4). In Chapter 2, two subspace based DOA estimation algorithms are briefly introduced, in which the root-MUSIC is better than the MUltiple Signal Classification (MUSIC) method because root-MUSIC does not suffer the grid error (and hence...
reduces point errors) while MUSIC does. In order to extend the root-MUSIC designed for uniform linear arrays (ULA) to arbitrary arrays, however, the cost of computational complexity is also increased. For the purpose of reducing the complexity of the extended root-MUSIC, two computationally efficient algorithms are proposed. In the first method, only a few largest eigenvalues corresponding to the desired roots are needed to calculate. The second method transforms the polynomial into a form of inverse discrete Fourier transform (IDFT) and thus the rooting process is replaced with the operation of scanning multiple circles in parallel. Both proposed methods achieve the same performance as the extended root-MUSIC, but with less computational burden.

Due to a host of practical reasons, the exact knowledge of the manifold of the desired signal is often unavailable. Chapter 3 introduces a model of the desired signal, in which the manifold is assumed to lie in a known linear subspace whereas how the bases of this linear subspace are combined to form the manifold is unknown. Two different linear subspaces are discussed which can accommodate pointing errors. It proves that the associated linear combination vector is able to be found by computing the principal eigenvector of a certain matrix which is designed without recourse to the interference subspace. Then a so-called matched direction beamformer is constructed, which is robust to pointing errors and overestimation of the signal subspace dimension.

In Chapter 4, the conventional Wiener-Hopf and the “modified” Wiener-Hopf beamformers are analyzed from the standpoint view of subspace. It is found that the Wiener-Hopf processors suffer from the problems of the power inversion and lacking robustness to pointing errors. The “modified” Wiener-Hopf beamformers overcome these two problems by removing the desired signal effects. However both processors allow the interferences to pass through. The proposed
method employs the technique of vector space projections to eliminate much of the pointing errors. Then the power of the desired signal is estimated in a one-step operation. The desired-signal-absent covariance matrix is formed by subtracting the effects of the desired signal from the desired-signal-present covariance matrix. Thus a weight vector orthogonal to the interferences can be formed. The proposed beamformer provides a unified solution to the problems of the power inversion, the robustness to pointing errors, and the complete interference cancellation.

Finally, in Chapter 5 the thesis is concluded and a list of the original contributions as well as an outlook on future research are provided.
Chapter 2

Fast root-MUSIC for Arbitrary Arrays

DIRECTION-OF-ARRIVAL (DOA) estimation is an ubiquitous task in many array signal processing applications. Among the classic DOA estimation techniques, maximum likelihood (ML) methods have the reputation of the best estimation performance because they asymptotically approach the Cramér-Rao lower bound [9]. However, their computational complexity is usually prohibitive since a multi-dimensional search is needed to find the global maximum of the likelihood function [10]. One solution to simplify the computational complexity of ML methods while maintaining the high-resolution DOA estimation ability is to use the methods called subspace methods. The MUltiple Signal Classification (MUSIC) algorithm, invented by Schmidt [11, 12], is the first and maybe the most popular subspace method. MUSIC searches a reduced parameter space and in turn can be implemented with much less complexity as compared to the ML methods [10]. However, MUSIC still requires a spectral search process, the computational task of which may be unaffordable for some real-time implementations [10, 13]. In order to further reduce the computational complex-
ity, the search-free DOA estimation methods have been developed, which aim to avoid the spectral search step. The estimation of signal parameters via rotational invariance techniques (ESPRIT), proposed in [14], employs two identical, translationally invariant subarrays to estimate DOA, where the property that the signal subspaces of the two subarrays are shift-invariant is used. However, it has been shown in [15] that ESPRIT is statistically less efficient than MUSIC. Additionally, ESPRIT seriously suffers from the calibration errors which results in the discrepancy between the subarrays and in turn degrades the estimation performance. Another search-free approach is the root-MUSIC, proposed in [16], where the DOA estimation problem is reformulated as a polynomial rooting problem. In comparison with MUSIC, root-MUSIC not only substantially reduces the computational complexity but also improves threshold performance [17]. Unfortunately, these search-free methods are applicable only to specific array geometries. For instance, ESPRIT requires the subarrays with shift-invariant structure, and root-MUSIC is designed for ULA or the non-uniform arrays whose sensors lie on a uniform grid.

By using the manifold separation [18–20] or Fourier Domain root-MUSIC techniques [13], root-MUSIC has been extended to arrays with arbitrary geometry but at the cost of increased computational complexity. In this chapter, two fast algorithms are presented to reduce the computational cost. The first proposed method is implemented in an iterative way where the Schur algorithm [21] is conducted to factorize the Laurent structured polynomial. Then Arnoldi iteration [22] is employed to compute only a few of the largest eigenvalues. This implies that a large number of the unwanted eigenvalues (or roots) are exempt from the calculation and therefore the computational complexity is reduced significantly. The second approach has a work-efficient parallel implementation, in
which inverse discrete Fourier transform (IDFT) is used and polynomial rooting is avoided. Both proposed methods, with less computational burden, asymptotically exhibit the same performance as the extended root-MUSIC, implying that they outperform the conventional MUSIC in terms of resolution ability.

2.1 Background of Subspace DOA Estimation

Let an array with $N$ omnidirectional sensors receive $M$ narrowband signals located in the far field. The $N \times 1$ array snapshot vector at time $t$ can be modeled as

$$\mathbf{x}(t) = \sum_{i=1}^{M} \mathbf{S}(\theta_i) \mathbf{m}_i(t) + \mathbf{n}(t)$$

(2.1)

where $\mathbf{n}(t) \in \mathbb{C}^N$ is a complex noise vector which corrupts the received signal. Here it assumes that the noise is zero-mean additive white Gaussian noise (AWGN) and uncorrelated from sensor to sensor. It is also assumed that the signals are uncorrelated with each other and with the noise. Using matrix notation, the received snapshot vector may be rewritten in a compact form

$$\mathbf{x}(t) = \mathbf{S}\mathbf{m}(t) + \mathbf{n}(t)$$

(2.2)

where $\mathbf{m}(t) = [m_1(t), \ldots, m_M(t)]^T \in \mathbb{C}^M$ stands for the vector of $M$ complex narrowband signal envelopes. The $N \times M$ matrix $\mathbf{S}$ has columns the manifold vectors of the $M$ signals, i.e.,

$$\mathbf{S} = [\mathbf{S}(\theta_1), \ldots, \mathbf{S}(\theta_M)]$$

(2.3)
which is assumed to be of full column rank meaning that \( N > M \). The second order statistics of the vector-signal \( x(t) \) is represented by the covariance matrix \( R_{xx} \)

\[
R_{xx} = \mathcal{E}\{x(t)x^H(t)\} = SR_{mm}S^H + \sigma_n^2 I_N
\]  

(2.4)

where \( \sigma_n^2 \) denotes the power of the AWGN noise. The covariance matrix of signals is represented by \( R_{mm} \) which, under the assumption of uncorrelated signals, is given by

\[
R_{mm} = \mathcal{E}\{m(t)m^H(t)\} = \begin{bmatrix}
\sigma_1^2, & 0, & \ldots, & 0 \\
0, & \sigma_2^2, & \ldots, & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0, & 0, & \ldots, & \sigma_M^2
\end{bmatrix}
\]  

(2.5)

where \( \sigma_i^2 = \mathcal{E}\{m_i^2(t)\} \) is the power of the \( i^{th} \) signals. Performing eigen-decomposition on \( R_{xx} \) gives

\[
R_{xx} = \sum_{i=1}^{N} \lambda_i E_i E_i^H
\]  

(2.6)

where the eigenvalues \( \{\lambda_i, i = 1, \ldots, N\} \) are arranged in nonascending order (i.e., \( \lambda_1 \geq \ldots \geq \lambda_N \)), and \( E_i \) is the eigenvector corresponding to \( \lambda_i \). Theoretically, the smallest \( N - M \) eigenvalues are equal to the noise power, i.e.,

\[
\lambda_{M+1} = \ldots = \lambda_N = \sigma_n^2
\]  

(2.7)

Then \( R_{xx} \) may be partitioned into two parts as follows

\[
R_{xx} = E_s (D_s + \sigma_n^2 I_M) E_s^H + \sigma_n^2 E_n E_n^H
\]  

(2.8)
where the diagonal matrix $\mathbb{D}_s$ is defined as
\[
\mathbb{D}_s = \begin{bmatrix}
\lambda_1 - \sigma_n^2, & 0, & \ldots, & 0 \\
0, & \lambda_2 - \sigma_n^2, & \ldots, & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0, & 0, & \ldots, & \lambda_M - \sigma_n^2
\end{bmatrix}
\] (2.9)

$\mathbb{E}_s$ and $\mathbb{E}_n$ contain, respectively, the eigenvectors associated with the largest $M$ eigenvalues (or the first $M$ dominant eigenvectors) and the remaining eigenvectors, i.e.,
\[
\mathbb{E}_s \triangleq [E_1, E_2, \ldots, E_M] \\
\mathbb{E}_n \triangleq [E_{M+1}, E_{M+2}, \ldots, E_N]
\] (2.10)

Also, $\mathbb{E}_s$ and $\mathbb{E}_n$ are referred to as signal- and noise-subspace eigenvectors.

It is well known that the subspace spanned by $\mathbb{E}_s$ is equal to that spanned by $\mathbb{S}$ [11,12]. Moreover, the signal subspace is orthogonal to the noise subspace. In mathematics, the relationship among these three subspace is expressed as follows
\[
\mathcal{L}[\mathbb{E}_s] = \mathcal{L}[\mathbb{S}] \perp \mathcal{L}[\mathbb{E}_n]
\] (2.11)

In practical applications, the exact covariance matrix $\mathbb{R}_{xx}$ is unavailable and its sample estimate below is used.
\[
\hat{\mathbb{R}}_{xx} = \frac{1}{L} \sum_{l=1}^{L} \varphi(t_l)\varphi(t_l)^H
\] (2.12)

where $\varphi(t_l)$ denotes the snapshot vector received at time $t_l$ and $L$ is the number of the snapshots. For this case, the mean of the $N - M$ smallest eigenvalues is
used to estimate $\sigma_n^2$, i.e.,

$$\widehat{\sigma}_n^2 = \frac{\sum_{i=M+1}^{N} \lambda_i^2}{N - M}$$

(2.13)

2.1.1 Classical MUSIC

Due to the orthogonality between the signal and noise subspace, the angles where the projection of the corresponding manifold onto the noise subspace are zero should be the DOAs of signals. By exploiting this property, the MUSIC searches the continuous array manifold vector $S(\theta)$ over the area of $\theta$ to find the $M$ minima of the following null-spectrum function

$$\xi(\theta) = S^H(\theta)E_nE_n^H S(\theta) = \|E_n^H S(\theta)\|^2$$

(2.14)

The major drawbacks associated with MUSIC are:

1. The estimation error due to “grid” based search may occur. For instance, if the array manifold is searched in the step of 1°, the signal located at 10.5° may be not found. This grid error (or quantization error) definitely causes pointing error of the manifold vector [23].

2. The computational complexity is high, particularly for the real-time applications, which can be explained by the following fact. For the purpose of avoiding grid error, the required number of search points has to be significantly large. In other words, the angular search grid has to be fine enough to avoid the quantization problem [23]. To obtain each search point, moreover, a matrix product of $E_n^H$ and $S(\theta)$ and the associated norm (see Eq.(2.14)) have to be computed.
2.1 Background of Subspace DOA Estimation

2.1.2 Root-MUSIC

Suppose the array used is a ULA array, then the associated $y$ and $z$ coordinates of array sensors become $r_y = r_z = 0$, and the $x$ coordinate is expressed as

$$r_x = [0, D, \ldots, D^{N-1}]^T$$  \hspace{1cm} (2.15)

where the first sensor is chosen as the reference point, and $D$ stands for the distance between the adjacent sensors which is usually half wavelength. Now the corresponding manifold vector can be written as follows

$$S(\theta) = \exp(-j\pi r_x \cos \theta) = [1, z, \ldots, z^{N-1}]^T$$  \hspace{1cm} (2.16)

where $z$ is defined as

$$z = \exp(-j\pi D \cos \theta)$$  \hspace{1cm} (2.17)

Clearly the manifold in Eq.(2.16) is Vandermonde structured. Using the fact that $z^* = 1/z$ (where $(\cdot)^*$ denotes the complex conjugate operation), the MUSIC null-spectrum function can be rewritten as follows

$$\xi(z) = S^H(\theta)E_nE_n^H S(\theta)$$

$$= \begin{bmatrix} 1, & \frac{1}{z}, & \ldots, & \frac{1}{z^{N-1}} \end{bmatrix} E_n E_n^H \begin{bmatrix} 1, & z, & \ldots, & z^{N-1} \end{bmatrix}^T$$

$$= \sum_{i=-N+1}^{N-1} b_i z^{-i}$$  \hspace{1cm} (2.18)

where the coefficient $b_i$ is the sum of the elements of the matrix $E_n E_n^H$ along the $i^{th}$ diagonal, i.e.,

$$b_i = \sum_{\forall m-n=i} [E_n E_n^H]_{m,n}$$  \hspace{1cm} (2.19)
where the notation \([A]_{m,n}\) denotes the \((m, n)^{th}\) entry of the matrix \(A\). Clearly, \(\xi(z)\) is a univariate polynomial with degree of \((2N - 2)\). The DOAs of the \(M\) signals are corresponding to the roots of \(\xi(z)\). Thus the DOA estimation is reduced to a root finding problem, in which DOAs can be obtained as the phase angles of the roots closest to the unit circle.

Instead of searching over \(\theta\), the root version MUSIC overcomes the quantization problem and high computational complexity aforementioned in the MUSIC. Another favorable advantage of root-MUSIC over classical MUSIC is that root-MUSIC is immune to radial errors [17]. As shown in Figure 2.1, if the error \(\Delta z_i\) is along the radial direction, there is no error in the DOA estimation (i.e., \(\Delta \theta_i = 0\)). Nevertheless, such radial error affects the MUSIC because MUSIC searches the unit circle only. In Figure 2.1, it is illustrated that two closely spaced roots \(\hat{z}_1\) and \(\hat{z}_2\), due to radial error, may correspond to a common point \(\hat{z}_3\) on the unit circle. In such case, the resulting MUSIC angular spectrum has only one peak which causes an apparent loss in resolution, and hence the point error occurs, no matter how fine the search grid is.
2.2 Extended root-MUSIC for Arbitrary Arrays

The conventional root-MUSIC is restricted to the ULAs or the arrays with sensors lying on a uniform grid. In this section, three major methods which make the root-MUSIC applicable to arrays of arbitrary geometry will be briefly introduced.

2.2.1 Interpolated root-MUSIC

The basic idea behind interpolated root-MUSIC [24–26] is to approximate the actual non-uniform array by means of a virtual ULA with \( N_V \) sensors, i.e.,

\[
S(\theta) \approx G_V S_V(\theta)
\]  

(2.20)
2.2 Extended root-MUSIC for Arbitrary Arrays

where $S_V(\theta)$ is the $N_V \times 1$ manifold vector of the virtual ULA. Note that the dimension of the virtual array manifold is less than that of the actual manifold, i.e., $N_V \leq N$. The “thin” matrix $G_V \in \mathbb{C}^{N \times N_V}$ is the interpolation matrix designed to minimize the interpolation error. Using the above approximation, the null-spectrum of interpolated root-MUSIC can be defined as

$$
\xi_{\text{intp}} = S^H_V(\theta)G^H_V E_n^HE_n^H G_V S_V(\theta) \tag{2.21}
$$

Apparentely, the above null-spectrum has similar form with the standard root-MUSIC null-spectrum (see Eq.(2.18)) and thus the conventional root-MUSIC algorithm can be applied on it.

The main problem of interpolation root-MUSIC is that the approximation in Eq.(2.20) often introduces significant mapping errors which may cause DOA estimation bias and excess variance [27]. Another problem is that the approximation is inaccurate for the whole angular field-of-view, which implies that the mapping matrix $G_V$ is dependent on angular sector. Hence in the practical implementation, the information that which sector the DOAs belong to has to be given or estimated. The next two approaches avoid these two problems.

2.2.2 Manifold Separation Technique (MST)

The $n^{th}$ element of the manifold vector, denoted by $[S(r, \theta)]_n$, can be explicitly written as a function of both array geometry and DOA as follows

$$
[S(r, \theta)]_n = \exp \left( -j\pi \frac{r}{\lambda_n} u(\theta) \right)
= \exp \left( -j\pi (x_n \cos \theta + y_n \sin \theta) \right)
= \exp \left( -j\pi \|r_n\| \cos(\psi_n - \theta) \right) \tag{2.22}
$$
where \((x_n, y_n)\) denotes the \(x-y\) coordinates of the \(n^{th}\) sensor in units of half wavelength, and \((\|r_n\|, \psi_n)\) represents the corresponding polar coordinates, which is given by

\[
\begin{align*}
\|r_n\| &= \sqrt{x_n^2 + y_n^2} \\
\psi_n &= \tan^{-1}\left(\frac{y_n}{x_n}\right)
\end{align*}
\] (2.23)

By using the Jacobi-Anger expansion [28], \([S(r, \theta)]_n\) can be written as

\[
[S(r, \theta)]_n = \sum_{q=-\infty}^{\infty} j^q J_q(-\pi \|r_n\|) \exp(jq(\psi_n - \theta))
\]

\[
= \sum_{q=-\infty}^{\infty} j^q J_q(-\pi \|r_n\|) \exp(jq\psi_n) \exp(-jq\theta)
\] (2.24)

Thus the manifold vector \(S(r, \theta)\) can be written as

\[
S(r, \theta) = \sum_{q=-\infty}^{\infty} j^q \begin{bmatrix}
J_q(-\pi \|r_1\|) \\
J_q(-\pi \|r_2\|) \\
\vdots \\
J_q(-\pi \|r_N\|)
\end{bmatrix} \odot \begin{bmatrix}
\exp(jq\psi_1) \\
\exp(jq\psi_2) \\
\vdots \\
\exp(jq\psi_N)
\end{bmatrix} \exp(-jq\theta)
\] (2.25)

where \(\odot\) denotes the operation of Hadamard product (or elementwise product).

By ignoring the terms with \(q\) greater than \(\pm Q\) (where \(Q \triangleq \frac{N_r-1}{2}\)), the manifold vector \(S(r, \theta)\) may be rewritten as

\[
S(r, \theta) = G(r)d(\theta) + \xi
\] (2.26)
where $\mathbb{G}(\mathbf{r}) \in \mathbb{C}^{N \times N_V}$, dependent on the array geometry only, is defined as

$$
\mathbb{G}(\mathbf{r}) = \begin{bmatrix}
\mathbb{G}_{-Q}, & \mathbb{G}_{-Q+1}, & \cdots, & \mathbb{G}_{Q-1}, & \mathbb{G}_{Q}
\end{bmatrix} \in \mathbb{C}^{N \times N_V} \quad (2.27)
$$

with $\mathbb{G}_q \in \mathbb{C}^N$ defined in Eq.(2.25). The Vandermonde structured vector $d(\theta) \in \mathbb{C}^{N_V}$ is dependent on the azimuth only and has the following form

$$
d(\theta) = \begin{bmatrix}
Z^{-Q}, & Z^{-Q+1}, & \cdots, & Z^{Q-1}, & Z^Q
\end{bmatrix}^T \in \mathbb{C}^{N_V} \quad (2.28)
$$

with

$$
Z = \exp(-j\theta) \quad (2.29)
$$

$\varepsilon$ represents the truncation error, which decays superexponentially as $N_V$ increases and tends to zero as $N_V \to \infty$ [18]. This means that the vector $\varepsilon$ can be safely neglected without generating significant modeling errors, provided that $N_V$ is sufficiently large (generally $N_V \gg N$). Note that only the azimuth angle $\theta \in [0^\circ, 360^\circ)$ is considered in this thesis. For the 2-D case (azimuth and elevation estimation problem), spherical harmonics is used to model $d(\theta, \phi)$ [18].

Under the condition that $N_V$ is large enough such that $\varepsilon$ can be safely neglected, a polynomial is constructed as follows

$$
\xi(Z) = \mathbb{G}^H(\mathbf{r}, \theta)\mathbb{E}_n\mathbb{E}_n^H\mathbb{S}(\mathbf{r}, \theta)
= \mathbb{d}^H(\theta)\mathbb{G}^H(\mathbf{r})\mathbb{E}_n\mathbb{E}_n^H\mathbb{G}(\mathbf{r})\mathbb{d}(\theta)
= \frac{1}{2\pi} \sum_{i=-(N_V-1)}^{N_V-1} b_i Z^{-i} \quad (2.30)
$$

where the coefficient $b_i$ is the sum of entries of $\mathbb{G}^H(\mathbf{r})\mathbb{E}_n\mathbb{E}_n^H\mathbb{G}(\mathbf{r}) \in \mathbb{C}^{N_V \times N_V}$ along
the \(i^{th}\) diagonal, i.e.,

\[
b_i = \sum_{\forall m-n=i} [G^H(r)E_n E_n^H G(r)]_{m,n}
\]

(2.31)

\(\xi(Z)\) is a polynomial with degree of \((2N_V - 2)\), meaning that there are \((2N_V - 2)\) roots. \(M\) roots closest to the unit circle should be selected among the \((2N_V - 2)\) roots and the phase angles of the \(M\) roots are utilized to estimate the DOAs.

### 2.2.3 Fourier Domain (FD) root-MUSIC

In [13], a competitive alternative to the MST, called Fourier Domain (FD) root-MUSIC, has been proposed. The MUSIC null spectrum function Eq.(2.14) is periodic in \(\theta\) with period \(2\pi\), because \(\overline{S}(\theta) = \overline{S}(\theta + 2\pi)\). Therefore, the null spectrum can be rewritten by Fourier series expansion as

\[
\xi(\theta) = \sum_{i=-\infty}^{\infty} b_i e^{j i \theta} = \sum_{i=-\infty}^{\infty} b_i Z^i
\]

(2.32)

with

\[
Z = \exp(j\theta)
\]

(2.33)

The Fourier coefficients are obtained by

\[
b_i = \frac{1}{2\pi} \int_{-\pi}^{\pi} \xi(\theta) Z^{-i} d\theta
\]

(2.34)

Here the null spectrum function \(\xi(\theta)\) over the angular domain may be analogous to the conventional signal function over time domain. Similar to MST, the function \(\xi(\theta)\) can be truncated to finite dimension as

\[
\xi(\theta) \approx \sum_{i=-(N_V-1)}^{N_V-1} b_i Z^i
\]

(2.35)
where

$$b_i \approx \frac{1}{2\pi} \sum_{k=-(N_V-1)}^{N_V-1} \xi(k\Delta\theta)e^{-jik\Delta\theta}$$

$$= \frac{1}{2\pi} \sum_{k=-(N_V-1)}^{N_V-1} \left\| \mathbf{E}_n^H S(k\Delta\theta) \right\|^2 e^{-jik\Delta\theta} \quad (2.36)$$

with

$$\Delta\theta = \frac{2\pi}{2N_V - 1} \quad (2.37)$$

The coefficients \(\{b_i\}\) can be expressed in a compact matrix form as follows

$$\mathbf{b} = \begin{bmatrix} b_{-(N_V-1)}, & b_{-(N_V-2)}, & \ldots, & b_{(N_V-1)} \end{bmatrix}^T = \mathbf{F}\mathbf{B} \quad (2.38)$$

where

$$\mathbf{F} = \frac{1}{2\pi} \begin{bmatrix} e^{-j(N_V-1)(N_V-1)\Delta\theta}, & e^{-j(N_V-1)(N_V-2)\Delta\theta}, & \ldots, & e^{j(N_V-1)(N_V-1)\Delta\theta} \\
\vdots & \vdots & \ddots & \vdots \\
e^{j(N_V-1)(N_V-1)\Delta\theta}, & e^{j(N_V-1)(N_V-2)\Delta\theta}, & \ldots, & e^{-j(N_V-1)(N_V-1)\Delta\theta} \end{bmatrix} \quad (2.39)$$

and

$$\mathbf{B} = \begin{bmatrix} \left\| \mathbf{E}_n^H S(-N_V\Delta\theta) \right\|^2 \\
\left\| \mathbf{E}_n^H S(-(N_V-2)\Delta\theta) \right\|^2 \\
\vdots \\
\left\| \mathbf{E}_n^H S((N_V-1)\Delta\theta) \right\|^2 \end{bmatrix} \in \mathbb{R}^{2N_V-1} \quad (2.40)$$

Now the FD root-MUSIC may be implemented via the following steps:

1. Compute \((2N_V-1)\) MUSIC spectral points \(\{\xi(k\Delta\theta), k = -(N_V - 1), \ldots, N_V - 1\}\).

2. Calculate the \((2N_V-1)\) Fourier coefficients \(\{b_i\}\) using Eq.\((2.36)\) where the
method of discrete Fourier transform (DFT) can be used.

3. Construct the polynomial of Eq. (2.35) whose degree is \(2N_V - 2\), using \(\{b_i\}\).

4. Apply the root-MUSIC algorithm on this polynomial to estimate the DOAs.

Compared with MST technique, the FD root-MUSIC has achieved two improvements. One is the computational complexity of finding the polynomial coefficients is slightly smaller than that of MST. Another is that the DOA estimation variance of the FD root-MUSIC is smaller than that of MST when the polynomial degree is quite small. If the polynomial degree is relatively large, both FD root-MUSIC and MST have the same estimation performance.

One common problem of the MST and FD root-MUSIC is that they need to solve a polynomial with high degree, since the value of \(N_V\) has to be sufficiently large to restrict the truncation errors to a wanted level. The requirement of computing all the \((2N_V - 2)\) roots of (2.30) (or (2.35)), coupled with the fact that \(N_V\) is significantly large, may render the two methods computationally expensive, particularly when the sensor number \(N\) is also very large. In [13] an IDFT-based method, named line-search root-MUSIC, has been proposed. Nevertheless, the resolution ability of this method is inferior to the root-based methods.

Next, two algorithms will be developed to reduce the complexity of rooting the polynomial with high degree. It is important to point out that the proposed algorithms are applicable to both MST and FD root-MUSIC. Moreover, the resolution ability of the proposed methods are asymptotically same as the extended root-MUSIC.
2.3 Iterative Fast root-MUSIC

Let us reexamine (2.30) and (2.35). Both of them are of Laurent polynomial [29]. Taking into account the Hermitian property of the matrix $G_H(r)E_n^H G(r)$, one obtains the coefficients $b_i = b_{-i}^*$. Furthermore, the null spectrum is non-negative because $\xi(\theta) = \|E_n^H S(\theta)\|^2 \geq 0$. Since the polynomial resulted from MST is similar with that from FD root-MUSIC, only the case of MST is discussed in the rest of this chapter. The polynomial of (2.30) can be factorized into two parts, which is supported by the following lemma.

**Lemma 1.1** [21]: Consider a Laurent polynomial $\xi(Z)$,

$$\xi(Z) = \sum_{i=-(N_V-1)}^{N_V-1} b_i Z^{-i}, \quad \text{with } b_i = b_{-i}^*$$

(2.41)

and such that it is non-negative on the unit circle, $\xi(e^{j\Omega}) \geq 0$. Then the canonical factorization of $\xi(Z)$ is given by

$$\xi(Z) = c_1 \xi_1(Z)\xi_1^*(1/Z^*)$$

(2.42)

where $c_1$ is a positive constant. The roots of $\xi(Z)$ appear in conjugate reciprocal pairs, i.e., if $Z_1$ is a root of $\xi(Z)$, then $(Z_1^{-1})^*$ is also a root.

**Proof**: Also see [21]

Clearly the polynomial $\xi(Z)$ in (2.30) meets the conditions of the above lemma, which means that $\xi(Z)$ can be factorized into two parts. Moreover, computing half of the roots (i.e., roots of $\xi_1(Z)$) is sufficient to find the roots of interest. To this end, a spectral factorization method should be employed. An excellent survey of spectral factorization methods has been provided in [21]. In these methods, a sequence of banded Hermitian Toeplitz matrices is formed as
follows

\[ T_0 = b_0, \quad T_1 = \begin{bmatrix} b_0, & b_{-1} \\ b_1, & b_0 \end{bmatrix}, \ldots, \quad T_k = \begin{bmatrix} b_0, & b_{-1}, & \ldots, & b_{-k} \\ b_1, & b_0, & \ldots, & b_{-k+1} \\ \vdots & \vdots & \ddots & \vdots \\ b_k, & b_{k-1}, & \ldots, & b_0 \end{bmatrix} \] (2.43)

Performing triangular factorization on \( T_k \) gives

\[ T_k = L_k D_k L_k^H \] (2.44)

where \( L_k \) represents a lower triangular matrix with unit diagonal entries and \( D_k \) is a positive diagonal matrix. As \( k \to \infty \), the last \( N_V \) entries of the last row of \( L_k \) tend exponentially fast to the coefficients of \( \xi_1(Z) \) [21]. To accomplish the triangular factorization, the computationally efficient Schur algorithm, in which the Toeplitz structure is exploited, is employed in this thesis. The Schur method can be implemented via the following steps:

1. Initialize a \((N_V \times 2)\) matrix \( \mathbb{B}_0 \), using \( b_i \) calculated from (2.31).

\[
\mathbb{B}_0 = \begin{bmatrix} b_0, & b_{-1} \\ b_{-1}, & b_{-2} \\ \vdots & \vdots \\ b_{-(N_V-2)}, & b_{-(N_V-1)} \\ b_{-(N_V-1)}, & 0 \end{bmatrix}
\] (2.45)

2. For \( k = 1, 2, \ldots \) until convergence, iterate the following steps
2.3 Iterative Fast root-MUSIC

(a) $\mathbb{B}_k = \mathbb{B}_{k-1} U_k$, where $U_k$ is a $(2 \times 2)$ matrix defined as

$$U_k = \frac{1}{\sqrt{1 - |\gamma|^2}} \begin{pmatrix} 1 & -\gamma \\ -\gamma^* & 1 \end{pmatrix}$$

(2.46)

with $\gamma = \left[ \mathbb{B}_{k-1} \right]_{1,2} / \left[ \mathbb{B}_{k-1} \right]_{1,1}$, i.e., the ratio of the two entries of the first row of $\mathbb{B}_{k-1}$.

(b) Shift up the second column of $\mathbb{B}_k$ by one element while keeping the first column unaltered.

(c) Test for convergence $\| \hat{b}_{1,k} - \hat{b}_{1,k-1} \| < \text{threshold}$, where $\hat{b}_{1,k}$ and $\hat{b}_{1,k-1}$ denote the first column of $\mathbb{B}_k$ and $\mathbb{B}_{k-1}$ respectively. If converged, go to (3), else return to (2.a).

3. The coefficients of $\xi_1(Z)$ are $\hat{b}_{1,k}$.

Now the polynomial factor $\xi_1(Z)$, which has all its roots on or inside the unit circle, is split from $\xi(Z)$. To find the roots, one can construct an unsymmetric companion matrix $\mathbb{M}$ whose eigenvalues correspond to the roots of $\xi_1(Z)$. Let the polynomial $\xi_1(Z) = C_0 + C_1 Z + C_2 Z^2 + \ldots + C_{Nv-1} Z^{Nv-1}$. Then the corresponding companion matrix is given by [22,30]

$$\mathbb{M} = \begin{bmatrix}
0, & 0, & \cdots, & 0, & -\frac{C_0}{C_{Nv-1}} \\
1, & 0, & \cdots, & 0, & -\frac{C_1}{C_{Nv-1}} \\
0, & 1, & \cdots, & 0, & -\frac{C_2}{C_{Nv-1}} \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
0, & 0, & \cdots, & 1, & -\frac{C_{Nv-2}}{C_{Nv-1}}
\end{bmatrix}$$

(2.47)

The eigenvalues of interest, corresponding to the DOAs, must be the $M$ largest ones of $\mathbb{M}$ because all roots of $\xi_1(Z)$ are within the unit circle. This implies that
only a few largest eigenvalues of $M$ are necessary. Power method [22] is a classic and the simplest algorithm to find the largest eigenvalue. However, the power iteration suffers from the slow convergence when the gap between the first and the second largest eigenvalues are not sufficiently wide [22, 31]. Also, only one eigenvalue (the largest one) can be computed by the power method. To overcome these two drawbacks, the methods of Lanczos iteration and Arnoldi iteration have been proposed to find the large eigenvalues of symmetric and unsymmetric matrices respectively. Here the Arnoldi iteration is employed because the companion matrix $M$ is an unsymmetric matrix. See Appendix 1.A for the introduction of Arnoldi iteration. Note that the function eigs.m of MATLAB has implemented Arnoldi iteration by using the ARPACK software package [32].

To summarize, the proposed fast iterative root-MUSIC algorithm for arbitrary arrays can be accomplished via the following steps:

1. Compute the sampling matrix $G(r)$. Note that this off-line process requires to be done only once for a given array.

2. Form the received data covariance matrix and perform eigen-decomposition to obtain the noise subspace vectors $E_n$. Then the coefficients of $\xi(Z)$ can be calculated using (2.31).

3. Perform fast spectral factorization on $\xi(Z)$ via Schur Algorithm to obtain the polynomial factor $\xi_1(Z)$ and the corresponding companion matrix $M$.

4. Apply Arnoldi iteration method to calculate the $M$ largest eigenvalues of $M$. Then DOAs can be estimated by the phase angles of these eigenvalues.
2.4 IDFT-based root-MUSIC

In practice, the roots of (2.30) (or (2.35)) are not exactly on the unit circle any more in the presence of perturbation, e.g., the effects of finite snapshots (finite observation interval). Even the slightest perturbation can result in the roots moving away from the unit circle (pp. 209 of [33]). Therefore, it is reasonable to model the roots as

\[ Z = \rho e^{-j\theta} \]  

where \( \rho \) denotes the radius. Following the work of [34], one can rewrite the polynomial in (2.30) explicitly dependent on \( \theta \) and \( \rho \) by substituting (2.48) into (2.30)

\[
\xi(\theta, \rho) = \frac{1}{2\pi} \sum_{i=-N_V-1}^{N_V-1} b_i \rho^{-i} e^{ji\theta} = \frac{1}{2\pi} \sum_{i=-N_V-1}^{N_V-1} B_i(\rho) e^{ji\theta} = \frac{1}{2\pi} e^{j(1-N_V)\theta} \sum_{k=1}^{2N_V-1} B_{k-N_V}(\rho) e^{j(k-1)\theta} = \frac{1}{2\pi} e^{j(1-N_V)\theta} \sum_{k=1}^{K} \overline{B}_k(\rho) e^{j(k-1)\theta} \]  

(2.49)

where \( B_i(\rho) \triangleq b_i \rho^{-i} \) and \( k \triangleq i + N_V \). The number \( K \), no less than \((2N_V - 1)\), takes the value of a power of 2. The coefficient \( \overline{B}_k(\rho) \) is defined as

\[
\overline{B}_k(\rho) = \begin{cases} 
  B_{k-N_V}(\rho), & \text{if } 1 \leq k \leq (2N_V - 1) \\
  0, & \text{if } (2N_V - 1) < k \leq K 
\end{cases} \]  

(2.50)
Then the magnitude of (2.49) is given by

\[ |\xi(\theta, \rho)| = \frac{1}{2\pi} \left| \sum_{k=1}^{K} B_k(\rho) e^{j(k-1)\theta} \right| \]  

(2.51)

and thus a 2-D spectrum is defined as

\[ P_{\text{IDFT}}(n, \rho) = \frac{1}{\xi(\rho, \frac{2\pi(n-1)}{K})} \]

\[ = \frac{2\pi/K}{\frac{1}{K} \sum_{k=1}^{K} B_k(\rho) e^{j\frac{2\pi}{K}(k-1)(n-1)}} \]  

(2.52)

where \( n = 1, 2, \ldots, K \). Interestingly, the denominator of the above has a typical \( K \)-point IDFT form, meaning that the 2-D spectrum can be computed directly by using IDFT for a given \( \rho \). Apparently the peaks of the 2-D spectrum correspond to the true DOAs because the roots of (2.49) are associated with the true DOAs. The DOAs can be estimated by the follows.

\[ \theta_m = \frac{2\pi(n_m - 1)}{K} \]  

(2.53)

where \( n_m \) corresponds to the index-\( n \) of the \( m^{th} \) peak of the 2-D spectrum \( P_{\text{IDFT}}(n, \rho) \).

Also, Eq.(2.52) can be rewritten in a compact form as

\[ P_{\text{IDFT}} = \mathbb{F} \mathbb{B} \in \mathbb{C}^{K \times L} \]  

(2.54)

where \( L \) is the total number of circles required to scan. \( \mathbb{F} \in \mathbb{C}^{K \times K} \) is an inverse Fourier transform matrix and \( \mathbb{B} \in \mathbb{C}^{K \times L} \) denotes a coefficient matrix.
elements of the three matrices in Eq.(2.54) are defined as follows

\[
[P_{\text{IDFT}}]_{n,\ell} = \frac{2\pi}{P_{\text{IDFT}}(n, \rho_{\ell})} 
\] (2.55a)

\[
[F]_{n,k} = e^{j\frac{2\pi}{K}(k-1)(n-1)} 
\] (2.55b)

\[
[B]_{n,\ell} = B_{n}(\rho_{\ell}) 
\] (2.55c)

Thus, finding peaks in Eq.(2.52) is equivalent to finding minima in Eq.(2.54).

In summary, the proposed IDFT-based root-MUSIC algorithm can be accomplished via the following steps:

1. Compute the matrix \( G(r) \) (off-line process).

2. Form the covariance matrix \( R_{xx} \) by using the observed snapshots and perform eigen-decomposition to obtain the noise subspace vectors \( E_{n} \). Then the coefficients \( \{b_{i}\} \) can be calculated using (2.31).

3. For \( \rho = 1, 1 - \Delta \rho, \ldots \) until \( \rho = 1 - (L - 1)\Delta \rho \), do the following steps to scan the spectra along \( L \) circles inside the unit circle. \( \Delta \rho \) determines the grid of the circles.

   (a) Calculate the \((2N_{V} - 1)\) coefficients \( B_{i}(\rho) = b_{i}\rho^{-i} \). Then the coefficients \( \overline{B}_{k}(\rho) \) can be obtained by padding \( B_{i}(\rho) \) with \((K - 2N_{V} + 1)\) zeros.

   (b) Perform \( K \)-point IDFT operation on \( \overline{B}_{k}(\rho) \) and invert the results to obtain the spectrum \( P_{\text{IDFT}}(n, \rho) \) in (2.52) for the current radius \( \rho \).

4. Identify the \( M \) peaks closest to the unit circle via 2-D search on the spectra.

5. Estimate DOAs by using (2.53).

If one scans the unit circle only, i.e., \( \rho \) is fixed at 1, then the proposed algorithm reduces to line-search root-MUSIC (LS-root-MUSIC) [13,34] similar to
2.5 Computational Complexity Analysis

that proposed in [13, 34]. LS-root-MUSIC is essentially identical to the conventional MUSIC, except that all spectral points of LS-root-MUSIC are calculated by one $K$-point IDFT operation while in MUSIC each point is obtained by a matrix multiplication. LS-root-MUSIC and MUSIC are under the assumption that, corresponding to each true DOA, there is a peak in the spectrum along the unit circle. This is a stronger assumption than distinct Z-plane roots because the root-based methods are insensitive to the radial errors [17]. Hence, the proposed IDFT-based method is expected to have better resolution ability than LS-root-MUSIC and MUSIC.

2.5 Computational Complexity Analysis

The complexity order of the rooting in the extended root-MUSIC is $O((2N_V - 2)^3)$, when assuming that eigenvalue-based methods are used for rooting. That is the roots are found by computing the eigenvalues of the corresponding $(2N_V - 2) \times (2N_V - 2)$ companion matrix. In the proposed iterative algorithm, the Schur algorithm requires $O(2N_V^2)$ operation for the spectral factorization [35–37]. The complexity of an Arnoldi iteration is roughly $O(M^2N_V)$ [38, 39]. Considering the fact that $M$ is far less than $N_V$, the complexity of Arnoldi iteration may be ignored compared with the Schur algorithm. Thus the proposed iterative algorithm reduces the complexity of rooting from $O((2N_V - 2)^3)$ to $O(2N_V^2)$.

The complexity order of LS-root-MUSIC is $O(K \log_2 K)$. Each loop of Step 3 of the proposed IDFT-based method takes $O(2N_V)$ operations to compute the coefficient $\overline{B}_k(\rho)$ and $O(K \log_2 K)$ operations for $K$-point IDFT. Thus the overall complexity order of the second proposed method is $O(L(2N_V + K \log_2 K))$. Therefore, the computational complexity of the two IDFT-based methods will not
essentially increase when $N_V$ becomes large (e.g., in the applications of large scale arrays), whereas the complexity of the extended root-MUSIC will suffer because it increases cubically with $N_V$. Note the line search or 2-D search for the two IDFT-based methods has different meaning than the search process in the conventional MUSIC method. In the MUSIC method, each spectral point needs an operation of matrix production. However, in both IDFT-based methods, all spectral points are obtained simultaneously by one IDFT operation. Furthermore, the IDFT operation dominates the computing time and hereby the complexity of search process is not included in the analysis here.

It is important to point out that the spectra of the $L$ circles can be computed simultaneously in parallel time $O(2N_V + K \log_2 K)$ if using $L$ processors, which is comparable to that of LS-root-MUSIC. Despite the possible increase in computational complexity, the proposed algorithms generally offer greater estimation accuracy than LS-root-MUSIC.

2.6 Simulation Studies

Assume $M = 2$ uncorrelated equally-powered signals impinge on a non-ULA array of $N = 5$ sensors. A randomly generated $x$-$y$ Cartesian coordinates of array sensors, in units of half-wavelengths, is shown below

$$ r = \begin{pmatrix} 0, & 0.9, & 1.7, & 2.6, & 2.5 \\ 0, & 0.3, & -0.1, & 0.3, & -0.5 \end{pmatrix} $$

This array geometry is fixed throughout all simulation runs. The column size of $G(r)$ is set to be $N_V = 99$, which provides the modeling error $\|\varepsilon\| \leq 10^{-15}$. The grid of the circles is $\Delta \rho = 0.01$. The IDFT length is $K = 2^{13}$ and MUSIC
search grid is $0.01^\circ$. In the following experiments, five methods are examined: conventional MUSIC, LS-root-MUSIC, the extended root-MUSIC, the proposed iterative fast root-MUSIC and the proposed IDFT-based root-MUSIC.

**Case 1:** In the first example, the true DOAs are $[90^\circ, 95^\circ]$ and the input SNR is 20dB. The theoretical covariance matrix $R_{xx}$ (see Eq.(2.4)) is used. Figure 2.2 shows the angular spectra of the MUSIC and LS-root-MUSIC between $50^\circ$ and $130^\circ$. Note that the field-of-view of this 2-D array is $[0^\circ, 360^\circ]$ despite only the spectrum over $[50^\circ, 130^\circ]$ being plotted.

![Figure 2.2](image)

**Figure 2.2:** The angular spectra of MUSIC and LS-root-MUSIC using theoretical $R_{xx}$. The true DOAs are $[90^\circ, 95^\circ]$. The input SNR=20dB.

It can be seen from Figure 2.2 that the two spectrum curves almost
overlap perfectly, which means that the two methods are essentially identical. The two peaks of the two methods are located, respectively, at $(90^\circ, 95^\circ)$ and $(89.96^\circ, 94.97^\circ)$. The estimation errors of LS-root-MUSIC is due to the quantization-error caused by limited angular grid step (here the grid is $360^\circ/2^{13} = 0.0439^\circ$).

The theoretical covariance matrix is also processed by the three root-MUSIC-type methods. In Figure 2.3, roots of the extended root-MUSIC are plotted, along with the two roots obtained by the proposed iterative fast root-MUSIC. Though only the roots with magnitude in the area $[0.5, 2]$ are selected for the extend root-MUSIC, it is enough to reveal the essence. The 2-D spectrum of the proposed IDFT-based method is depicted in Figure 2.4 with $\rho \in [0.5, 1]$ and $\Delta \rho = 0.01$ (i.e., $L=51$). Note that in practice $L$ is not required to be such a large number because empirically $\rho \in [0.9, 1]$ is sufficient to locate the true DOAs. Our aim here is to give a more comprehensive view of the spectrum of the proposed method. Figure 2.3 and 2.4 demonstrate that the three root-MUSIC-type approaches can find the two DOAs as well.

In order to further examine the simulation results, the roots of the extended root-MUSIC and the proposed iterative method, in conjunction with the $(\rho, \theta)$ pairs of the peaks of the two IDFT-based methods, are listed in Table 2.1 and 2.2 respectively. As shown in Table 2.1, the roots of the extended root-MUSIC satisfy the conjugate reciprocity property. That is the roots outside and inside the unit circle have the common phase angle but reciprocal magnitudes. The roots computed by the proposed iterative method approximate ideally the roots of interest by the extended root-MUSIC. In Table 2.2, it can be seen that the peaks of LS-root-MUSIC are corresponding to the proposed IDFT-based method with $\rho = 1$. Importantly, the $(\rho, \theta)$ pairs of peaks of the proposed IDFT-based
method are in close proximity to the roots of the extended root-MUSIC within the unit circle. Therefore, these three root-MUSIC-type approached obtain the same roots of interest.
Figure 2.4: The 2-D spectrum of the proposed IDFT-based method (and contour diagram), where $\rho \in [0.5, 1]$ and $\Delta \rho = 0.01$. The theoretical $R_{xx}$ is used. The true DOAs are $[90^\circ, 95^\circ]$. The input SNR=20dB.
Table 2.1: The roots of the extended root-MUSIC and the proposed iterative fast root-MUSIC when the theoretical covariance matrix $R_{xx}$ is used.

<table>
<thead>
<tr>
<th>extended root-MUSIC</th>
<th>proposed iterative fast root-MUSIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>$-0.8098 - 1.5692i = 1.77e^{-j117.30^\circ\pi/180^\circ}$</td>
<td>$0.0011 - 0.9904i = 0.99e^{-j89.94^\circ\pi/180^\circ}$</td>
</tr>
<tr>
<td>$1.2076 + 0.9941i = 1.56e^{-j120.54^\circ\pi/180^\circ}$</td>
<td>$-0.0874 - 0.9865i = 0.99e^{-j95.06^\circ\pi/180^\circ}$</td>
</tr>
<tr>
<td>$1.3726 - 0.4439i = 1.44e^{-j117.92^\circ\pi/180^\circ}$</td>
<td>$0.0000 - 1.0000i = 1.00e^{-j90^\circ\pi/180^\circ}$</td>
</tr>
<tr>
<td>$0.2046 + 1.3545i = 1.37e^{-j278.59^\circ\pi/180^\circ}$</td>
<td>$-0.0872 - 0.9962i = 1.00e^{-j185^\circ\pi/180^\circ}$</td>
</tr>
<tr>
<td>$-0.8648 + 0.8359i = 1.20e^{-j224.03^\circ\pi/180^\circ}$</td>
<td>$0.0000 - 1.0000i = 1.00e^{-j90^\circ\pi/180^\circ}$</td>
</tr>
<tr>
<td>$-0.0872 - 0.9962i = 1.00e^{-j185^\circ\pi/180^\circ}$</td>
<td>$0.0000 - 1.0000i = 1.00e^{-j90^\circ\pi/180^\circ}$</td>
</tr>
<tr>
<td>$0.0000 - 1.0000i = 1.00e^{-j90^\circ\pi/180^\circ}$</td>
<td>$-0.0874 - 0.9865i = 0.99e^{-j95.06^\circ\pi/180^\circ}$</td>
</tr>
<tr>
<td>$-0.0872 - 0.9962i = 1.00e^{-j185^\circ\pi/180^\circ}$</td>
<td>$0.0000 - 1.0000i = 1.00e^{-j90^\circ\pi/180^\circ}$</td>
</tr>
<tr>
<td>$-0.5978 + 0.5778i = 0.83e^{-j224.03^\circ\pi/180^\circ}$</td>
<td>$0.0000 - 1.0000i = 1.00e^{-j90^\circ\pi/180^\circ}$</td>
</tr>
<tr>
<td>$0.1090 + 0.7218i = 0.73e^{-j278.59^\circ\pi/180^\circ}$</td>
<td>$-0.0874 - 0.9865i = 0.99e^{-j95.06^\circ\pi/180^\circ}$</td>
</tr>
<tr>
<td>$0.6596 - 0.2133i = 0.69e^{-j117.92^\circ\pi/180^\circ}$</td>
<td>$0.0000 - 1.0000i = 1.00e^{-j90^\circ\pi/180^\circ}$</td>
</tr>
<tr>
<td>$0.4936 + 0.4063i = 0.64e^{-j224.03^\circ\pi/180^\circ}$</td>
<td>$0.0000 - 1.0000i = 1.00e^{-j90^\circ\pi/180^\circ}$</td>
</tr>
<tr>
<td>$-0.2597 - 0.5033i = 0.51e^{-j117.30^\circ\pi/180^\circ}$</td>
<td>$0.0000 - 1.0000i = 1.00e^{-j90^\circ\pi/180^\circ}$</td>
</tr>
</tbody>
</table>

Table 2.2: $(\rho, \theta)$ pairs of the peaks of LS-root-MUSIC and the proposed IDFT-based method when the theoretical covariance matrix $R_{xx}$ is used.

<table>
<thead>
<tr>
<th>LS-root-MUSIC</th>
<th>proposed IDFT-based method</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1.00, 89.96°)</td>
<td>(1.00, 89.96°)</td>
</tr>
<tr>
<td>(1.00, 94.97°)</td>
<td>(1.00, 94.97°)</td>
</tr>
<tr>
<td>(0.83, 223.99°)</td>
<td>(0.83, 223.99°)</td>
</tr>
<tr>
<td>(0.73, 278.53°)</td>
<td>(0.73, 278.53°)</td>
</tr>
<tr>
<td>(0.69, 17.89°)</td>
<td>(0.69, 17.89°)</td>
</tr>
<tr>
<td>(0.64, 320.49°)</td>
<td>(0.64, 320.49°)</td>
</tr>
<tr>
<td>(0.57, 117.25°)</td>
<td>(0.57, 117.25°)</td>
</tr>
</tbody>
</table>
Case 2: In the second example, the covariance matrix $\mathbf{R}_{xx}$ is formed by using Eq.(2.12) where 100 snapshots are obtained from one Monte-Carlo realization with DOAs=$[90^\circ, 95^\circ]$ and input SNR=20dB. The spectra of the two MUSIC-type methods are illustrated in Figure 2.5 in which only one peak appears. This means that the two methods fail to distinguish the two incoming signals. The roots and spectrum of the three root-MUSIC-type methods are depicted in Figure 2.6 and 2.7. The detailed data are listed in Table 2.3 and 2.4. It is clear that the three root-MUSIC-type algorithms have better resolution ability than the former two MUSIC-type methods in this example because two DOAs are provided by the
latter three methods, which can be explained by the fact that the root-MUSIC-type methods are insensitive to the radial errors [17]. Also, it can be found in Table 2.3 and 2.4 that the three root-MUSIC-type approaches yield the same roots of interest. One important observation is that the radii corresponding to the true DOAs are not exactly equal to unit anymore. This is due to the perturbation caused by the covariance matrix estimation using finite snapshots. In practice, the perturbation is unavoidable. Therefore it is more sensible to find the DOAs on the circles with radius rather than unit.

![Figure 2.6: Roots for the extended root-MUSIC and the proposed iterative fast root-MUSIC.](image)

Figure 2.6: Roots for the extended root-MUSIC and the proposed iterative fast root-MUSIC. $R_{xx}$ is formed by using 100 snapshots of one realization. The true DOAs are $[90^\circ, 95^\circ]$. The input SNR=20dB.
Figure 2.7: The 2-D spectrum of the proposed IDFT-based method (and contour diagram), where $\rho \in [0.5, 1]$ and $\Delta \rho = 0.01$. $R_{xx}$ is formed by using 100 snapshots of one realization. The true DOAs are $[90^\circ, 95^\circ]$. The input SNR = 20dB.
2.6 Simulation Studies

Table 2.3: The roots of the extended root-MUSIC and the proposed iterative fast root-MUSIC when the covariance matrix $R_{xx}$ is formed by using 100 snapshots of one realization.

<table>
<thead>
<tr>
<th>extended root-MUSIC</th>
<th>proposed iterative fast root-MUSIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>$1.5987 - 0.8297i = 1.80e^{-j27.43^\circ \pi/180^\circ}$</td>
<td>$-0.0128 - 0.9702i = 0.97e^{-j90.75^\circ \pi/180^\circ}$</td>
</tr>
<tr>
<td>$-1.2747 + 0.9778i = 1.61e^{-j217.49^\circ \pi/180^\circ}$</td>
<td>$-0.0735 - 0.9657i = 0.97e^{-j94.35^\circ \pi/180^\circ}$</td>
</tr>
<tr>
<td>$0.8488 + 1.1989i = 1.47e^{-j305.39^\circ \pi/180^\circ}$</td>
<td>$-0.0721 + 0.6945i = 0.70e^{-j264.07^\circ \pi/180^\circ}$</td>
</tr>
<tr>
<td>$-0.1479 + 1.4246i = 1.43e^{-j264.07^\circ \pi/180^\circ}$</td>
<td>$0.3934 + 0.5556i = 0.68e^{-j305.39^\circ \pi/180^\circ}$</td>
</tr>
<tr>
<td>$-0.0784 - 1.0296i = 1.03e^{-j94.35^\circ \pi/180^\circ}$</td>
<td>$-0.4939 + 0.3788i = 0.62e^{-j217.49^\circ \pi/180^\circ}$</td>
</tr>
<tr>
<td>$-0.0136 - 1.0305i = 1.03e^{-j90.75^\circ \pi/180^\circ}$</td>
<td>$0.4928 - 0.2557i = 0.56e^{-j27.43^\circ \pi/180^\circ}$</td>
</tr>
</tbody>
</table>

Table 2.4: $(\rho, \theta)$ pairs of the peaks of LS-root-MUSIC and the proposed IDFT-based method when the covariance matrix $R_{xx}$ is formed by using 100 snapshots of one realization.

<table>
<thead>
<tr>
<th>LS-root-MUSIC</th>
<th>proposed IDFT-based method</th>
</tr>
</thead>
<tbody>
<tr>
<td>$ (1.00, 91.76^\circ)$</td>
<td>$ (0.97, 90.70^\circ)$</td>
</tr>
<tr>
<td></td>
<td>$ (0.97, 94.31^\circ)$</td>
</tr>
<tr>
<td></td>
<td>$ (0.70, 264.02^\circ)$</td>
</tr>
<tr>
<td></td>
<td>$ (0.68, 305.24^\circ)$</td>
</tr>
<tr>
<td></td>
<td>$ (0.62, 217.44^\circ)$</td>
</tr>
<tr>
<td></td>
<td>$ (0.56, 27.38^\circ)$</td>
</tr>
</tbody>
</table>

In the following three examples, the root-mean-square-error (RMSE) performance of DOA estimation is concerned. For each scenario, the average of 500 independent Monte-Carlo runs is used to obtain each simulated point.

**Case 3:** In the third example, the DOA of the second signal source varies from $91^\circ$ to $100^\circ$ whereas the DOA of the first signal is fixed at $90^\circ$. All other parameters are chosen as the same as that in the second example. In Figure 2.8, the DOA estimation RMSEs versus the signal angular separation are presented.
Figure 2.8: DOA estimation RMSEs versus $(\theta_2 - \theta_1)$ with the snapshot number = 100, input SNR=20dB, $\theta_1 = 90^\circ$, $\rho \in [0.9, 1]$ and $\Delta \rho = 0.01$.

Case 4: In the fourth example, the performances with different numbers of snapshots are investigated while all other parameters remain as in the second example. The DOA estimation RMSEs versus the snapshot number are displayed in Figure 2.9.
Case 5: The last example studies the impact of input SNR on the five methods tested. Also, all other parameters are the same as in the second example. The DOA estimation RMSEs versus input SNR are plotted in Figure 2.10.

All the three figures demonstrate that the proposed two algorithms provide the asymptotically similar performance in DOA estimation to the extended root-MUSIC. Furthermore, as expected, the proposed algorithms have superior capability to MUSIC-type methods in the situations: when two signal sources are closely spaced, when the snapshot number is quite small, or when input SNR is low.
2.7 Summary

Two computationally efficient root-MUSIC algorithms for arbitrary arrays have been presented in this chapter. The extended root-MUSIC has made it possible to apply the traditional root-MUSIC to the arrays with arbitrary geometry. However, the problem that the extended root-MUSIC has to face is that a polynomial with very high order is required to root, which may be computationally expensive.

Figure 2.10: DOA estimation RMSEs versus the input SNR with DOAs=[90°, 95°], the snapshot number =100, $\rho \in [0.9, 1]$ and $\Delta \rho = 0.01$. 
Considering the facts that the roots appear in conjugate reciprocal pairs and the roots of interest are closest to the unit circle, the first proposed iterative method suggests a framework in which, rather than computing all roots, only the wanted roots need to be computed. It basically consists of a combination of the spectral factorization and large eigenvalue finding. The polynomial is efficiently split, via the Schur algorithm, into two factors with roots, respectively, inside and outside the unit circle. The Schur algorithm exploits the Toeplitz structure to complete triangular factorization. The desired roots are corresponding to a few large roots of the new polynomial with roots inside the unit circle. Then Arnoldi iteration is utilized to compute only the large eigenvalues of the associated companion matrix.

The second proposed method is an IDFT-based root-MUSIC in which the DOAs are obtained by scanning a range of circles. The essence behind is that due to the inevitable perturbation the roots corresponding to the true DOAs are no longer located on the unit circle and hence multiple circles need to be concerned. The second proposed algorithm is computationally efficient as IDFT is adopted and it is readily to scan all circles in parallel.

The analysis and simulation results verify that the proposed algorithms, with less computational burden, have the asymptotically similar performance in DOA estimation to the extended root-MUSIC. Also, the proposed algorithms have superior resolution ability to LS-root-MUSIC (which is a MUSIC type algorithm) and the conventional MUSIC, particularly when two signal sources are located close to each other in space, the number of snapshots is relatively small, or the input SNR is low.
2.8 Appendix 2.A Arnoldi Iteration

Starting with an random non-zero vector \( \mathbf{v}_0 \), every power iteration produces a unity norm vector \( \mathbf{v}_k \) as follows [22]

\[
\mathbf{v}_k = \frac{A \mathbf{v}_{k-1}}{\|A \mathbf{v}_{k-1}\|}
\]  

(2.56)

where \( A \) is the square matrix whose eigenvalues are wanted. The sequence \( \{\mathbf{v}_k, k = 1, 2, \ldots\} \) converges to the principal eigenvector after \( K - 1 \) iterations and thus the principal eigenvalue can be obtained. However, only the final result, \( A^{K-1} \mathbf{v}_0 \), is used. The successive vectors, \( A^{K-M} \mathbf{v}_0, \ldots, A^{K-2} \mathbf{v}_0, A^{K-1} \mathbf{v}_0 \) (where \( M \) denotes the number of the largest eigenvalues required to be computed), which may contain potentially useful information, are ignored by the power iteration.

Another drawback of the power iteration method is that only one eigenvalue (the principal eigenvalue) is obtained after \( K - 1 \) iterations. These two problems can be addressed by the method of Arnoldi Iteration.

The additional eigen-information may be extracted by the Krylov subspace methods. The Krylov subspace is defined as [32]

\[
\mathcal{K}(A, \mathbf{v}, M) = \mathcal{L}[\mathbf{v}, A \mathbf{v}, A^2 \mathbf{v}, \ldots, A^{M-1} \mathbf{v}]
\]  

(2.57)

In general, these \( M \) vectors are not orthogonal each other. Gram-Schmidt orthogonalization may be applied to find an orthogonal basis of the Krylov subspace, represented by \( Q_M = [q_1, q_2, \ldots, q_M] \). Then orthogonal projection of \( A \) onto the Krylov subspace gives

\[
H_M = Q_M^H A Q_M
\]  

(2.58)

where \( H_M \in \mathbb{C}^{M \times M} \) is of Hessenberg form [22]. Therefore, the eigenvalues of \( H_M \)
can be computed efficiently. Also, the $M$ eigenvalues are supposed to converge to the $M$ largest eigenvalues of $A$. Here only rudiments of Arnoldi Iteration are presented. For further details, see [31,32].
Chapter 3

Matched Direction Beamforming based on Signal Subspace

In this chapter, the problem of constructing the matched direction beamformer (MDB) is studied, without using the interference subspace. MDB is referred to as the beamformer resolving the desired signal that is drawn from an unknown direction inside a known subspace [40, 41]. Due to many practical reasons, the exact knowledge of the DOA of the desired signal is often unavailable, which causes the mismatch between the actual and the nominal manifold vectors. For instance, the DOA estimator may be lacking in accuracy. In this chapter, the manifold vector of the desired signal in the presence of pointing errors is assumed to lie in a known linear subspace, but it remains unknown. The main contribution of this chapter is to propose a new MDB that can find the manifold associated with the desired signal without the knowledge of the interference subspace. The proposed approach is based on searching for the principal eigenvector of a certain rank-one matrix consisting of the signal subspace and the subspace which the desired signal belongs to. Furthermore, the proposed MDB is robust to both pointing errors and the dimension overestimation of the signal subspace.
3.1 Received Signal Model

Consider an array with $N$ sensors collects one desired signal and $M$ interference signals. The $(M + 1)$ signals are assumed to be narrow-band, uncorrelated to each other and located in the far field. The received signal vector is given by

$$x(t) = S(\theta_d)m_d(t) + \sum_{i=1}^{M} S(\theta_i)m_i(t) + n(t)$$

(3.1)

where the subscript $(\cdot)_d$ stands for the desired signal. $\theta_i$ and $m_i(t)$ represent, respectively, the DOA and complex envelope of the $i^{th}$ interference signal. The DOA of the desired signal, $\theta_d$, is often not perfectly known in practical applications. This “pointing” errors can result in the difference between the actual manifold and the presumed (nominal) manifold. That is

$$\text{if } \theta_d \neq \theta_0, \text{ then } S(\theta_d) \neq S(\theta_0)$$

(3.2)

where $\theta_0$ is the presumed DOA used by the array processor.

The straightforward consequence of this mismatch in DOA is to cause a substantial performance degradation of conventional adaptive beamformers (for further details, see the next chapter or [8]). As $\theta_d$ is unknown, the corresponding manifold vector $S(\theta_d)$ remains unknown. However, $S(\theta_d)$ lies in a known $p$-dimension (where $p > 1$) linear subspace $\mathcal{L}[\mathbb{H}]$ and thus can be written as a linear combination of the bases of $\mathcal{L}[\mathbb{H}]$ [40,41]. This means that the manifold vector $S(\theta_d)$ can be expressed as

$$S(\theta_d) = \mathbb{H}b$$

(3.3)

where the vector $b$ is referred to as the linear combination vector. Note that $b$
and \( \mathcal{S}(\theta_d) \) are unknown in practical situation, while \( \mathbb{H} \) is a known matrix. Next, two examples will be presented to demonstrate how to obtain the matrix \( \mathbb{H} \).

In the first example, the actual manifold is expanded in Taylor series around the presumed manifold \( \mathcal{S}(\theta_0) \) as \([40, 42, 43]\)

\[
\mathcal{S}(\theta_d) = \mathcal{S}(\theta_0 + \Delta\theta_d) = \mathcal{S}(\theta_0) + \sum_{k=1}^{\infty} \frac{(\Delta\theta_d)^k}{k} \left. \frac{\partial^k \mathcal{S}(\theta)}{\partial \theta^k} \right|_{\theta=\theta_0} (3.4)
\]

with

\[
\Delta\theta_d = \theta_d - \theta_0 \tag{3.5}
\]

where the nominal DOA, \( \theta_0 \), is known. In practice the first two order derivatives are enough to satisfy the wanted precision. For the details of the expressions for the first two order derivatives, see Appendix 3.A. Hence a matrix can be defined as

\[
\mathbb{H}_1 = [\mathcal{S}_0, \dot{\mathcal{S}}(\theta_0), \ddot{\mathcal{S}}(\theta_0)] \tag{3.6}
\]

where \( \dot{\mathcal{S}}(\theta_0) \) and \( \ddot{\mathcal{S}}(\theta_0) \) denote the first and second derivatives with respect to \( \theta_0 \). \( \mathbb{H}_1 \) can be computed prior to the beamformer processing if the presumed \( \theta_0 \) and array geometry information are at hand. Therefore it is reasonable to assume that \( \mathbb{H}_1 \) can be formed using the nominal direction. In the case when the array reference point is chosen as the centroid point of the array, the columns of \( \mathbb{H}_1 \) are orthogonal to each other [2]. Then the manifold of interest may be approximated as

\[
\mathcal{S}(\theta_d) \approx \mathbb{H}_1 \begin{bmatrix} 1 \\ \Delta\theta_d \\ \frac{(\Delta\theta_d)^2}{2} \\ \Delta \end{bmatrix} \tag{3.7}
\]

where \( \Delta \) is unknown.
Alternatively, using the Taylor series expansion and retaining the terms up to the second order, one has the following approximations

\[
S(\theta_0) = S(\theta_d - \Delta \theta_d) \\
\approx S(\theta_d) - \Delta \theta_d \dot{S}(\theta_d) + \frac{(\Delta \theta_d)^2}{2} \ddot{S}(\theta_d)
\]

\[
S(\theta_0 \pm \Delta \theta) = S(\theta_d \pm \Delta \theta - \Delta \theta_d) \\
\approx S(\theta_d) + (\pm \Delta \theta - \Delta \theta_d) \dot{S}(\theta_d) + \frac{(\pm \Delta \theta - \Delta \theta_d)^2}{2} \ddot{S}(\theta_d)
\]

where \(\Delta \theta\) is related to the expected range of the DOA of the desired signal and can be a given value even if the true DOA is unavailable. This means that \(\Delta \theta\) is known. Generally the value of \(\Delta \theta\) is set such that \(|\Delta \theta \pm \Delta \theta_d|\) is insignificant so that the above approximations are valid. Now \(S(\theta_d)\) may be rewritten as

\[
S(\theta_d) \approx \frac{(\Delta \theta)^2 - (\Delta \theta_d)^2}{(\Delta \theta)^2} S(\theta_0) + \frac{(\Delta \theta_d)^2 + \Delta \theta \Delta \theta_d}{2(\Delta \theta)^2} S(\theta_0 + \Delta \theta) \\
+ \frac{(\Delta \theta_d)^2 - \Delta \theta \Delta \theta_d}{2(\Delta \theta)^2} S(\theta_0 - \Delta \theta)
\]

\[
= \begin{bmatrix}
S(\theta_0), S(\theta_0 + \Delta \theta), S(\theta_0 - \Delta \theta)
\end{bmatrix}
\begin{bmatrix}
(\Delta \theta)^2 - (\Delta \theta_d)^2 \\
(\Delta \theta_d)^2 + \Delta \theta \Delta \theta_d \\
(\Delta \theta_d)^2 - \Delta \theta \Delta \theta_d
\end{bmatrix}
\begin{bmatrix}
S(\theta_0) \\
S(\theta_0 + \Delta \theta) \\
S(\theta_0 - \Delta \theta)
\end{bmatrix}
\]

Hence \(S(\theta_d)\) also belongs to the known linear subspace \(\mathcal{L}[S(\theta_0), S(\theta_0 - \Delta \theta), S(\theta_0 + \Delta \theta)]\) and the associated linear combination vector \(b_2\) is unknown.

In [44–46], an uncertainty model called flat ellipsoidal uncertainty set has been discussed in which the true manifold vector is expressed as

\[
S(\theta_d) = S(\theta_0) + \mathbb{B}_2 b_0, \quad \|b_0\| \leq 1
\]
3.1 Received Signal Model

where $\mathbb{B}$ is a known $N \times (p - 1)$ matrix with full column rank, and the vector $b_{fe}$ is unknown. Clearly the actual manifold can be rewritten as

$$S(\theta_d) = \mathbb{H}[1, b_{fe}^T]^T$$

(3.11)

where $\mathbb{H} \triangleq [S(\theta_0), \mathbb{B}]$. Therefore the flat ellipsoidal uncertainty set can be transformed to the linear subspace model of (3.3).

In this thesis, the unknown vector $b$ is assumed to stay unchanged during the observation interval of $L$ successive snapshots although it may vary in the next $L$ snapshots. The contribution of the desired signal to the covariance matrix $\mathbb{R}_{xx}$ is given by

$$\mathbb{R}_{dd} = \mathcal{E}\{(m_d(t)S(\theta_d))(m_d(t)S(\theta_d))^H\}$$

$$= \mathcal{E}\{m_d(t)m_d^*(t)\}S(\theta_d)S^H(\theta_d)$$

$$= \sigma_d^2 \mathbb{H} b b^H \mathbb{H}^H$$

(3.12)

which is a rank-1 matrix obviously. Under the assumption that the $M$ interference signals are uncorrelated with the desired signal, the second-order model for $\mathbf{x}(t)$ can be written as

$$\mathbb{R}_{xx} = \sigma_d^2 \mathbb{H} b b^H \mathbb{H}^H + \mathbb{R}_i + \sigma_n^2 \mathbb{I}$$

(3.13)

where the rank-$M$ matrix $\mathbb{R}_i$ represents the interference effects and may have the following form

$$\mathbb{R}_i = \sum_{i=1}^{M} \sigma_i^2 S(\theta_i)S^H(\theta_i)$$

(3.14)

The so-called matched direction beamforming (MDB) is referred to as the beamformer resolving the signal that are drawn from an unknown direction $\hat{b}$ inside the known subspace $\mathcal{L}[\mathbb{H}]$ [40, 41]. The main task of this chapter is to
design a MDB which is robust to pointing errors.

3.2 Previous Work

Two related existing approaches regarding MDB will be briefly introduced. These are:

1. MDB with the knowledge of interference subspace

2. Multirank minimum-variance-distionless-response (MVDR) beamformer

3.2.1 MDB with the Knowledge of Interference Subspace

Let $\mathcal{S}_I$ represent the matrix consisting of the interference manifold vectors. That is

$$\mathcal{S}_I = [\mathcal{S}(\theta_1), \ldots, \mathcal{S}(\theta_M)] \quad (3.15)$$

In [40], it assumes that $\mathcal{S}_I$ is exactly known and the authors derive a maximum-likelihood estimator of the vector $\mathcal{b}$, given by

$$\mathcal{b} = \beta \mathcal{P}\{(\mathbf{U}^H \mathbf{U})^{-1} \mathbf{U}^H \mathbf{R}_{xx} \mathbf{U}\} \quad (3.16)$$

where the notation $\mathcal{P}\{\cdot\}$ stands for the principal eigenvector of the matrix argument between braces, and the matrix $\mathbf{U}$ is defined as

$$\mathbf{U} = \mathbf{P}^\perp_{\mathcal{S}_I} \mathbf{H} \quad (3.17)$$

with

$$\mathbf{P}^\perp_{\mathcal{S}_I} = \mathbb{I} - \mathcal{S}_I (\mathcal{S}_I^H \mathcal{S}_I)^{-1} \mathcal{S}_I^H \quad (3.18)$$
The coefficient $\beta$ is a normalization constant. Note that the array beamformer output SNIR is independent of $\beta$. Finally the weight vector of the MDB is expressed as

$$w = \beta \mathcal{P}_{\delta_i} \mathcal{H} b = \beta U b$$  \hspace{1cm} (3.19)

where the coefficient $\beta$ can be determined such that $\|w^H w\| = 1$. The above beamformer intends to cancel the interferences completely while being matched to the best direction in the interference-free signal subspace.

The problem in [40] is that the perfect knowledge of the interference subspace is required for its implementation. In many practical applications, unfortunately, this assumption is not always valid or the interference subspace cannot be estimated easily.

### 3.2.2 Multirank MVDR Beamformer

In [41], a multirank minimum-variance-distionless-response (MVDR) beamformer is presented. This beamformer may be implemented by using a generalized side-lobe canceller (GSC) [47], illustrated in Figure 3.1. In the GSC, the input array signal vector $\mathcal{X}(t) \in \mathbb{C}^N$ is firstly decomposed into two parts: $\mathcal{H}^H \mathcal{X}(t) \in \mathbb{C}^p$ and $\mathcal{Q}^H \mathcal{X}(t) \in \mathbb{C}^{(N-p)}$. Note that here the columns of $\mathcal{H}$ are required to be orthonormal columns, i.e., $\mathcal{H}^H \mathcal{H} = I_p$. For the general $\mathcal{H}$ the Gram-Schmidt algorithm [22] may be employed to render the columns of $\mathcal{H}$ orthonormal. $\mathcal{Q} \in \mathbb{C}^{N \times (N-p)}$ is a left-orthogonal matrix that makes $[\mathcal{H} \mathcal{Q}]$ unitary, i.e., $[\mathcal{H} \mathcal{Q}][\mathcal{H} \mathcal{Q}]^H = I_N$. 
The middle branch of the GSC passes the desired signal and part of interferences and noise that lie in $\mathcal{L}[H]$. The bottom branch is the sidelobe canceling path where the desired signal is blocked. Then the residual signal $\varepsilon(t) \in \mathcal{C}^p$ is formed by

$$
\varepsilon(t) = H^H \tilde{x}(t) - F^H Q^H \tilde{x}(t)
$$

where $F^H = H^H R_{xx} Q (Q^H R_{xx} Q)^{-1}$ is designed by using the criterion of the linear minimum mean-squared error (LMMSE) so that the power of $\varepsilon(t)$ is minimized. The second-order statistics of $\varepsilon(t)$ is represented by the matrix $R_{ee}$

$$
R_{ee} = \mathcal{E}\{\varepsilon(t)\varepsilon^H(t)\} = H^H R_{xx} H - H^H R_{xx} Q (Q^H R_{xx} Q)^{-1} Q^H R_{xx} H \quad (3.21)
$$
Alternatively, it is shown in [41] that $R_{ee}$ has a simplified form as
\[
R_{ee} = (\mathbf{H}^H R_x^{-1} \mathbf{H})^{-1}
\] (3.22)

Finally, the desired signal is extracted from $e(t)$ by a weight matrix $\mathbf{W}$. The key idea of the multirank MVDR beamformer is how to design the weight matrix $\mathbf{W}$. Let us start from the simple case where only one interference is present. Then the matrix $R_{ee}$ may be expressed as
\[
R_{ee} = \sigma_d^2 b b^H + \eta \mathbf{H}^H \mathbf{S}(\theta_1) S^H(\theta_1) \mathbf{H} + \sigma_n^2 \mathbf{I}
\] (3.23)

where $\mathbf{S}(\theta_1)$ is the manifold vector of the interference signal. $\eta$ represents the level how the interference is suppressed by the GSC beamformer, which is given by [41]
\[
\eta = \frac{\sigma_d^2}{\sigma_d^2 + \sigma_n^2} \frac{\mathbf{S}(\theta_1)}{\mathbf{P}^\perp \mathbf{H} \mathbf{S}(\theta_1)}
\] (3.24)

It is clear that $b$ may be approximated by the principal eigenvector of $R_{ee}$ up to a scaling factor if the desired signal subspace $\mathcal{L}[\mathbf{H}]$ and the interference subspace $\mathcal{L}[\mathbf{S}(\theta_1)]$ are well separated. However, when the interference subspace is relatively close to $\mathcal{L}[\mathbf{H}]$, the effects of interferences on $R_{ee}$ cannot be neglected any more. The immediate consequence is that the principal eigenvector contains significant contributions from the interference. On the other hand, the desired signal contributes not only to the principal eigenvector but also to the other eigenvectors. In order to deal with this situation, [41] suggests that, instead of one eigenvector, a set of dominant eigenvectors of $R_{ee}$ (i.e., the eigenvectors corresponding to a few largest eigenvalues of $R_{ee}$), say $k$ dominant eigenvectors, should be utilized to restore the desired signal. This is the so-called multirank MVDR beamforming.

Thus the weight matrix of the multirank MVDR for the general case may
be explicitly expressed as

\[ W = R_{xx}^{-1} HR_{ee} J \]  

(3.25)

where \( J \) is composed of \( k \) dominant eigenvectors of \( R_{ee} \). In addition, the inequalities \( k \leq M + 1 \leq p \) must hold for the multirank MVDR. Finally, these multiple outputs are summed up by multiplication with the \( k \times 1 \) unit vector \( 1_k \) to obtain the final output. The array output SNIR can be written as

\[
SNIR_{\text{out}} = \frac{\text{tr}\{W^H R_{dd} W\}}{\text{tr}\{W^H (R_i + \sigma_n^2 I) W\}}
\]  

(3.26)

The major drawbacks associated with the multirank MVDR beamformer are:

1. each column of \( W \) is contaminated by the interferences, particularly when the interferences are close to the desired signal, which results in the degradation of array output SNIR;

2. the signal number is required to be not greater than the dimension of \( \mathbb{H} \), i.e., \( M + 1 \leq q \); hence this method fails in the case where a large number of interferences are present;

3. it is not clear how many dominant eigenvectors (i.e., the value of \( k \)) should be selected to build \( J \).

\section*{3.3 Proposed Matched Direction Beamformer}

\subsection*{3.3.1 Estimate using Signal Subspace}

Performing eigen-decomposition on \( R_{xx} \) yields:

\[
R_{xx} = \sum_{i=1}^{N} \lambda_i E_i E_i^H
\]  

(3.27)
3.3 Proposed Matched Direction Beamformer

where the eigenvalues \( \{ \lambda_i, i = 1, \ldots, N \} \) are the eigenvalues arranged in decreasing order (i.e., \( \lambda_1 \geq \ldots \geq \lambda_N \)). \( E_i \) is the eigenvector associated with \( \lambda_i \).

Inspired by [40], here an algorithm is developed, which is able to estimate the linear combination vector \( b \) without the knowledge of the interference subspace. The key point of this algorithm is supported by the following lemma.

**Lemma 3.1:** The vector \( b \) is the principal eigenvector of a rank-1 matrix \( Z_\ell \) up to a scaling factor, where the matrix \( Z_\ell \) is defined as

\[
Z_\ell = (U_\ell^H U_\ell)^{-1} U_\ell^H E_\ell E_\ell^H U_\ell
\]  

with

\[
U_\ell = P_{\mathcal{H}} \mathcal{H}
\]

The matrix \( \mathcal{H}_\ell \) is composed of all the \( M + 1 \) signal eigenvectors except \( E_\ell \), i.e.,

\[
\mathcal{H}_\ell = [E_1, \ldots, E_{\ell-1}, E_{\ell+1}, \ldots, E_{M+1}]
\]

\( E_\ell \) can be any \( M + 1 \) signal eigenvector.

**Proof:** \( E_\ell \) can be written as:

\[
E_\ell = E_\ell (E_\ell^H E_\ell)^{-1} \frac{E_\ell^H S(\theta_d)}{E_\ell^H S(\theta_d)} = \frac{E_\ell (E_\ell^H E_\ell)^{-1}}{E_\ell^H S(\theta_d)} \frac{E_\ell^H S(\theta_d)}{E_\ell^H S(\theta_d)} = \frac{P_{E_\ell} S(\theta_d)}{E_\ell^H S(\theta_d)}
\]

where the fact \( (E_\ell^H E_\ell)^{-1} = 1 \) is used in the above. \( E_\ell \) lies in the signal-subspace, implying that \( E_\ell \) is the combinations of the manifolds of the desired signal and all interferences. Hence \( E_\ell^H S(\theta_d) \neq 0 \) holds. Inserting (3.31) into \( Z_\ell \) defined in
(3.28) yields
\[
Z_\ell = \frac{1}{E_\ell^H S(\theta_d)} (U_\ell^H U_\ell)^{-1} U_\ell^H \mathbb{P}_E S(\theta_d) E_\ell^H U_\ell = \hat{\zeta}
\] (3.32)

Let \( \mathbb{P}_n \) denote the projection onto the noise subspace \( \mathcal{L}[E_{M+2}, \ldots, E_N] \). Then one has the followings
\[
\mathbb{P}_n S(\theta_d) = 0
\]
\[
\mathbb{P}_E + \mathbb{P}_{\mathbb{A}_\ell} + \mathbb{P}_n = \mathbb{I}
\] (3.33)

Consequently, the term \( \zeta \) can be rewritten as
\[
\zeta = U_\ell^H (\mathbb{I} - \mathbb{P}_{\mathbb{A}_\ell} - \mathbb{P}_n) S(\theta_d)
\]
\[
\zeta = U_\ell^H S(\theta_d) - U_\ell^H \mathbb{P}_{\mathbb{A}_\ell} S(\theta_d) - U_\ell^H \mathbb{P}_n S(\theta_d)
\]
\[
\zeta = U_\ell^H S(\theta_d) - H H^H \mathbb{P}_{\mathbb{A}_\ell} S(\theta_d) = 0
\] (3.34)

where the Hermitian property of projection operator \( \mathbb{P}_{\mathbb{A}_\ell} = (\mathbb{P}_{\mathbb{A}_\ell})^H \) is used. Replacing the term \( \zeta \) in (3.32) by (3.34) gives
\[
Z_\ell = \frac{1}{E_\ell^H S(\theta_d)} (U_\ell^H U_\ell)^{-1} U_\ell^H \mathbb{P}_E S(\theta_d) E_\ell^H U_\ell
\]
\[
Z_\ell = \frac{1}{E_\ell^H S(\theta_d)} (U_\ell^H U_\ell)^{-1} U_\ell^H H^H \mathbb{P}_{\mathbb{A}_\ell} \mathbb{P}_{\mathbb{A}_\ell}^+ U_\ell E_\ell^H U_\ell
\]
\[
Z_\ell = \frac{1}{E_\ell^H S(\theta_d)} (U_\ell^H U_\ell)^{-1} U_\ell^H \mathbb{P}_{\mathbb{A}_\ell}^+ \mathbb{P}_{\mathbb{A}_\ell} \mathbb{P}_{\mathbb{A}_\ell}^+ U_\ell E_\ell^H U_\ell
\]
\[
Z_\ell = \frac{1}{E_\ell^H S(\theta_d)} (U_\ell^H U_\ell)^{-1} U_\ell^H E_\ell^H U_\ell
\]
\[
Z_\ell = \frac{1}{E_\ell^H S(\theta_d)} b E_\ell^H U_\ell
\] (3.35)

where the idempotent property of projection operator \( \mathbb{P}_{\mathbb{A}_\ell}^+ = \mathbb{P}_{\mathbb{A}_\ell} \mathbb{P}_{\mathbb{A}_\ell}^+ \), the Hermi-


3.3 Proposed Matched Direction Beamformer

tian property of projection operator \( P_{A_{\ell}} \) = \((P_{A_{\ell}})^H\), and the modelling \( S(\theta_d) = Hb \) (see Eq. (3.3)) are utilized. Finally multiplying both sides of (3.35) by \( b \) produces

\[
Z_{\ell}b = \left( \frac{E_{\ell}^H U_{\ell} b}{E_{\ell}^H S(\theta_d)} \right) b
\]

(3.36)

It is obvious that the vector \( b \) is the principal eigenvector of the rank-1 matrix \( Z_{\ell} \) up to a scaling factor. □

In practice, the covariance matrix \( R_{xx} \) is estimated by Eq. (2.12). If the number of snapshots acquired by the array is quite small, the orthogonality between \( \hat{P}_n \) and \( S(\theta_d) \) may be impaired due to the effect of finite snapshot, particularly when the magnitudes of eigenvalues corresponding to the noise subspace are small. In such case, the term \( \zeta \) becomes \( \zeta = U_{\ell}^H S(\theta_d) - U_{\ell}^H \hat{P}_n S(\theta_d) \). Thus the estimated principal eigenvector and the corresponding manifold become

\[
\hat{b}^{(\ell)} = \beta b - \beta (U_{\ell}^H U_{\ell})^{-1} U_{\ell}^H \hat{P}_n S(\theta_d)
\]

\[
\hat{S}^{(\ell)}(\theta_d) = H\hat{b}^{(\ell)} = \beta S(\theta_d) - \beta H (U_{\ell}^H U_{\ell})^{-1} U_{\ell}^H \hat{P}_n S(\theta_d)
\]

(3.37)

where \( \beta \) is determined such that \( \| \hat{S}^{(\ell)}(\theta_d) \|^2 = N \). With the increasing of snapshot number, \( \hat{P}_n S(\theta_d) \) and the estimation error both tend to zero. Keeping in mind that the estimates in (3.37) are obtained from the matrix \( Z_{\ell} \) where the \( \ell^{th} \) signal-subspace eigenvector is excluded from \( A_{\ell} \), one can find \((M + 1)\) estimates of \( b \) if \( \{ Z_{\ell}, \ell = 1, 2, \ldots, M+1 \} \) are used. For the purpose of estimation error reduction, the proposed method employs the average of these \((M + 1)\) estimates instead of
3.3 Proposed Matched Direction Beamformer

one estimate, i.e.,

\[
\hat{b} = \frac{1}{M + 1} \sum_{\ell=1}^{M+1} \hat{b}^{(\ell)}
\]

\[
\hat{S}(\theta_d) = \mathbb{H} \hat{b} = \mathbb{H} \left( \frac{1}{M + 1} \sum_{\ell=1}^{M+1} \hat{b}^{(\ell)} \right)
\] (3.38)

The weight vector of the proposed MDB can be constructed as

\[
w_{prop} = \hat{R}_{xx}^{-1} \mathbb{H} \hat{b} = \frac{\hat{R}_{xx}^{-1} \mathbb{H}}{M + 1} \sum_{\ell=1}^{M+1} \hat{b}^{(\ell)}
\] (3.39)

In summary, the proposed MDB algorithm consists of the following steps.

1. Estimate the covariance matrix \( \hat{R}_{xx} \) by using the received snapshots and perform eigen-decomposition on \( \hat{R}_{xx} \) to obtain \( M + 1 \) signal-subspace eigenvectors \( \{E_i, i = 1, \cdots, M + 1\} \).

2. For \( \ell = 1, \cdots, M + 1 \), form the matrix \( Z_\ell \) in (3.28) and compute the associated principal eigenvector to obtain the vector \( \hat{b}^{(\ell)} \).

3. Average these \( M + 1 \) vectors \( \{\hat{b}^{(\ell)}, \ell = 1, \cdots, M + 1\} \) to obtain \( \hat{b} \) (see (3.38)).

4. Use (3.39) to compute the weight vector of the proposed MDB beamformer.

3.3.2 Some Discussions

The assumption made in Lemma 3.1 is that signal subspace eigenvectors \( [E_1, \cdots, E_{M+1}] \) are available, which is a quite weak assumption compared with the assumption in [40] where it assumes the interference subspace, \( \mathcal{L}[S_I] \), is perfectly known.

Consider the case when the signal subspace dimension \( M + 1 \) is overes-
timated. Then the dimensionality of the noise subspace is reduced. However, the projector $P_n$ with reduced dimensionality is still orthogonal to $S(\theta_d)$, which implies that Lemma 3.1 is valid in this case as well. Therefore the proposed method is robust to the overestimation of signal subspace. If the signal subspace dimension is underestimated, the estimated noise subspace would contain signal component which renders $P_n$ not orthogonal with $S(\theta_d)$ any more. Thus the proposed method fails in the case of dimension underestimation.

In [41, 48], the situation where the vector $b$ is not constant but random during the $L$ snapshot observation time due to the fast varying environment has been considered. In this case, the contribution due to the desired signal is given by

$$R_{dd} = a_d^2 HR_{bb} H$$

where $R_{bb} = E\{bb^H\}$ is a full rank matrix (rank-$p$). Therefore, $R_{dd}$ becomes a rank-$p$ matrix. In [41] only the simplest case where the rank-$p$ matrix $R_{bb}$ is assumed to be known is studied. Using Corollary VI.2 of [48] one can estimate the rank-$p$ matrix associated with the desired signal even if the structure of the rank-$p$ matrix is unknown. However, this approach cannot be employed in the scenario of this thesis. This is because that when using Corollary VI.2 of [48], $R_{bb}$ is required to be full rank such that its inversion, $R_{bb}^{-1}$, exists. Nevertheless, in the situation of this thesis the matrix $R_{bb} = bb^H$ is of rank-one and hereby non-invertible.

### 3.4 Simulation Studies

In order to evaluate the effectiveness of the proposed MDB, a number of simulation studies have been carried out using, without any loss of generality, a uniform
linear array with \( N = 10 \) sensors and half-wavelength sensor spacing. Note that the proposed method is applicable to arrays with arbitrary geometries. The array operates in the presence of three equally powered source signals where one is the desired signal and two are interferences. It is assumed a 3° pointing error in the desired signal direction with the actual DOA \( \theta_d = 90^\circ \) and the nominal DOA \( \theta_0 = 87^\circ \). The matrix \( H \) is obtained by using Eq.(3.6).

**Case 1:** In the first example, the powers of three signals are all unit and the noise power is \( \sigma^2_n = 0.1 \). The first interference signal is fixed at 100° while the DOA of the second interference varies from 67° to 86°. The covariance matrix \( R_{xx} \) has the theoretical form as Eq.(3.13). Four methods below have been simulated:

1. the multirank MVDR
2. the traditional Wiener-Hopf
3. the MDB proposed in [40] (see Eq.(3.19)) where the perfect knowledge of the interference subspace (IS) is assumed to be known
4. the proposed method

The weight vector of the Wiener-Hopf processor is given by [5]

\[
\mathbf{w}_{WH} = R_{xx}^{-1} S(\theta_0)
\]  

(3.41)

The SNIR performances of these methods versus the DOA of the second interference are displayed in Figure 3.2 where the legend ‘Multirank MVDR-\( k \)’ means that \( k \) dominant eigenvectors of \( R_{ee} \) are used to form the weight matrix in Eq.(3.25).
3.4 Simulation Studies

![Figure 3.2: Array output SNIR versus the azimuth of the 2nd interference. The 1st interference remains at 100°. The actual DOA of the desired signal is 90° while the presumed DOA is 87°.](image)

It can be seen from Figure 3.2 that the multirank MVDR with only the principal eigenvector of $R_{ee}$ is better than that with two or three dominant eigenvectors. That is because the second and third dominant eigenvectors contain significant effects of the interferences. When the second interference is far away from the desired signal (the DOA of the second interference is less than 79° in this example), MDB with perfect knowledge of the interference subspace, the multirank MVDR-1 and the proposed MDB have similar performance. Compared with the multirank MVDR, Figure 3.2 shows that the second interference has to be closer to the direction of the desired signal before it leads to a SNIR deterioration for the proposed method. The superiority of the MDB in [40] over the proposed method is mainly due to the assumption used in [40] that the exact
knowledge of the interference subspace is available at all times. Note that this assumption, however, is often impractical in many situations. The SNIR of the proposed algorithm begins to degrade when the DOA of the second interference is greater than $81^\circ$. That is because the second interference, as well as the desired signal, lies in the subspace $\mathcal{L}[H]$ if the DOA of the second interference is very close to the nominal DOA. Figure 3.2 also illustrates that the conventional Wiener-Hopf is quite susceptible to the pointing errors in this example.

**Case 2:** All the parameters in the second example are the same as that in the first example, except that the DOA of the first interference moves closer to the desired signal (which changes from $100^\circ$ to $97^\circ$). Figure 3.3 indicates that there exists a big gap between the multirank MVDR and the proposed method as well as the MDB in [40] even when the second interference is far away from the desired signal. This is because the first interference and the desired signal are closer in space than the first example, and thus the principal eigenvector of $R_{ee}$ contains significant component of the first interference. However, the proposed method maintains its performance.

**Case 3:** In this example, the scenario is identical to the first example, but the DOA of the second interference is fixed at $80^\circ$ and the input SNR varies between -10dB and 10dB. Figure 3.4 demonstrates that proposed method achieves the similar array output SNIR performance as that of the MDB using the exact knowledge of interference subspace. Also, as depicted in Figure 3.4, the multirank MVDR fails when input SNR is lower than -2dB because the noise dominates the principal eigenvector of $R_{ee}$ if the noise power is too strong.
3.4 Simulation Studies

**Figure 3.3:** Array output SNIR versus the azimuth of the 2nd interference. The 1st interference remains at $97^\circ$. The actual DOA of the desired signal is $90^\circ$ while the presumed DOA is $87^\circ$.

**Figure 3.4:** Array output SNIR versus the input SNR. The two interference DOAs are at $[80^\circ, 100^\circ]$. The actual DOA of the desired signal is $90^\circ$ while the presumed DOA is $87^\circ$. All signals are equally powered.
Case 4: The robustness of the proposed algorithm to the overestimation or underestimation of the signal subspace dimension is examined in the fourth example, with the same simulation scenario as the first example except that the second interference is fixed at $80^\circ$. The nominal signal subspace dimension is changed from 1 to 9, while the actual dimension is always 3 (i.e., $M + 1 = 3$). Figure 3.5 illustrates that the SNIR performance of the proposed method maintains when the nominal dimension is between 3 and 8. The proposed approach fails when the nominal dimension is underestimated (the nominal dimension is 1 or 2) because the estimated $P_n$ contains the signal component which leads to $P_n S(\theta_d) \neq 0$. When the nominal dimension is above 8, the proposed approach also fails which can be explained by the following fact. If the rank of the matrix $A_\ell$ in Eq.(3.30) is over 8, then the rank of $P_{A_\ell}^\perp$ is $N - 8 = 2$. Thus the rank of the matrix product $U_\ell = P_{A_\ell}^\perp H$ is not greater than 2. Therefore the $3 \times 3$ matrix $U_\ell H U_\ell^H$ becomes rank deficient and non-invertible.

Case 5: The effect of finite snapshots is investigated in this example, where the covariance matrix $R_{xx}$ is estimated by using Eq.(2.12). All other parameters are the same as the previous example but with the correct signal subspace dimension. The average of 500 independent simulation runs is used to plot each simulation point in Figure 3.6. Also, Figure 3.6 indicates that the proposed method is better than the multirank MVDR.
3.4 Simulation Studies

Figure 3.5: Array output SNIR versus the nominal signal subspace dimension. The true dimension is 3. The actual DOA of the desired signal is $90^\circ$ while the presumed DOA is $87^\circ$. The DOAs of interferences are $[80^\circ, 100^\circ]$.

Figure 3.6: Array output SNIR versus the snapshot number. The actual DOA of the desired signal is $90^\circ$ while the presumed DOA is $87^\circ$. The DOAs of interferences are $[80^\circ, 100^\circ]$. 
Case 6: Finally the estimations performance using one or all \( \{Z_\ell, \ell = 1, 2, \ldots, M + 1 \} \) are examined, with the same simulation environment as the previous example. The root-mean-square-error (RMSE) of the desired manifold versus the number of snapshots is plotted in Figure 3.7, where 500 independent Monte Carlo runs are performed. It can be seen that when the snapshot number is small, the estimate in (3.38) using all \( \{Z_\ell, \ell = 1, 2, \ldots, M + 1 \} \) is better than that using only one \( Z_\ell \).

![Figure 3.7: RMSE of the desired manifold versus the snapshot number. The interference signals are from \([80^\circ, 100^\circ]\). The actual DOA of the desired signal is 90° while the presumed DOA is 87°.](image)

3.5 Summary

In many practical applications the pointing errors may occur, due to, for instance, the DOA estimation errors. In such case the presumed manifold associated with
the desired signal used by the array system is different than the actual one. The consequence of this discrepancy is the substantial performance degradation for the conventional adaptive array beamformers (e.g., the traditional Wiener-Hopf beamformer), which can be seen in the first and second simulation examples.

In this chapter, the unknown actual manifold vector is modelled as a vector lying in a known subspace. Two different expressions of this subspace are also discussed, in which the Taylor series expansion is employed. Thus the true manifold can be approximated by a product of a known matrix and an unknown vector. This vector is also called linear combination vector by means of which the actual manifold can be obtained by linearly combining the subspace bases.

The main contribution of this chapter is that a novel algorithm is developed to estimate the linear combination vector without the knowledge of the interference subspace. This means that the condition with the perfect knowledge of interference subspace in [40] may be relaxed. Instead, the proposed estimator takes use of the signal subspace which can be easily obtained by performing eigen-decomposition on the covariance matrix. Then a Wiener-Hopf type MDB is proposed. The theoretical analysis and simulations have proven that the proposed method is robust to both pointing errors and the dimension overestimation of the signal subspace. Simulations also reveal that the performance of the proposed MDB outperforms the multirank MVDR in the case where the interferences are relatively close to the desired signal or the noise power is quite strong.
3.6 Appendix 3.A The first two derivatives of the manifold vector

This appendix presents the expression of first two derivatives of the manifold vector with respect to azimuth. The manifold vector may be expressed as follows

\[ S(\theta) = \exp[-j\pi(r_x \cos \theta + r_y \sin \theta)] \quad (3.42) \]

The first derivative is readily obtained as

\[ \dot{S}(\theta) = \frac{\partial S(\theta)}{\partial \theta} \quad (3.43) \]

\[ = \text{diag}(j\pi(r_x \sin \theta - r_y \cos \theta))S(\theta) \]

\[ = j\pi(r_x \sin \theta - r_y \cos \theta) \odot S(\theta) \]

where \( \odot \) denotes the operation of Hadamard product (or elementwise product).

The second derivative is written as

\[ \ddot{S}(\theta) = \frac{\partial^2 S(\theta)}{\partial \theta^2} \quad (3.44) \]

\[ = \text{diag}(j\pi(r_x \cos \theta + r_y \sin \theta))S(\theta) + \left(\text{diag}(j\pi(r_x \sin \theta - r_y \cos \theta))\right)^2 S(\theta) \]

\[ = j\pi \left( (r_x \cos \theta + r_y \sin \theta) + j\pi(r_x \sin \theta - r_y \cos \theta) \odot (r_x \sin \theta - r_y \cos \theta) \right) \odot S(\theta) \]
Chapter 4

Interference Cancellation
Beamforming Robust to Pointing Errors

An important topic concerned in array signal processing and wireless communications is interference (jammers) rejection. This can be achieved using a beamformer which places relatively high gain in the direction of the desired signal and nulls in the directions of the interferences.

The Wiener-Hopf processor suffers from the power inversion problem and has substantial performance degradation in the presence of pointing errors [8]. By removing the effects of the desired signal, the “modified” Wiener-Hopf avoids these two problems [49] but allows the interferences to pass through as well.

In this chapter, a novel array processor is presented which not only prevents power inversion and reduces the effect of pointing error but also provides, asymptotically, complete interference rejection. To eliminate much of the pointing error effects, the vector space projections (VSP) method is utilized to find a new manifold vector. The power of the desired signal can be estimated in one
step computation. Thus the effects of the desired signal can be excised so as to form the desired-signal-absent covariance matrix. Then a weight vector orthogonal with the interference subspace can be constructed. Numerical results demonstrate the superior performance of the proposed beamformer relative to other existing approaches.

### 4.1 Introduction and Literature Review

Consider an array of $N$ sensors receiving $M + 1$ uncorrelated narrowband signals (one desired signal and $M$ interferences) located in the far-field. The received $N \times 1$ signal-vector $\mathbf{x}(t)$ and its second order statistics are given by

$$
\mathbf{x}(t) = \mathbf{S}_d \mathbf{m}_d(t) + \sum_{i=1}^{M} \mathbf{S}_i \mathbf{m}_i(t) + \mathbf{n}(t)
$$

$$
\mathbf{R}_{xx} = \mathcal{E}\{\mathbf{x}(t)\mathbf{x}^H(t)\}
= \sigma_d^2 \mathbf{S}_d \mathbf{S}_d^H + \sum_{i=1}^{M} \sigma_i^2 \mathbf{S}_i \mathbf{S}_i^H + \sigma_n^2 \mathbf{I}
\triangleq \mathbf{R}_{i+n}
$$

(4.1)

where $\mathbf{S}_d$ and $\mathbf{S}_i$ stand for the manifold vectors of the desired signal and the $i^{th}$ interference respectively. Performing eigen-decomposition on $\mathbf{R}_{xx}$ produces

$$
\mathbf{R}_{xx} = \sum_{i=1}^{N} \lambda_i \mathbf{E}_i \mathbf{E}_i^H
$$

$$
= \mathbf{E}_s (\mathbf{D}_s + \sigma_n^2 \mathbf{I}_{M+1}) \mathbf{E}_s^H + \sigma_n^2 \mathbf{E}_n \mathbf{E}_n^H
$$

(4.2)
where the eigenvalues \( \{ \lambda_i, i = 1, \ldots, N \} \) are arranged in decreasing order, the \((M + 1) \times (M + 1)\) diagonal matrix \( \mathbb{D}_s \) is given by

\[
\mathbb{D}_s = \begin{bmatrix}
\lambda_1 - \sigma_n^2 & 0 & \cdots & 0 \\
0 & \lambda_2 - \sigma_n^2 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & \lambda_{M+1} - \sigma_n^2
\end{bmatrix}
\] (4.3)

\( \mathbb{E}_s \) and \( \mathbb{E}_n \) contain, respectively, the \( M + 1 \) dominant eigenvectors and the remaining eigenvectors, i.e.,

\[
\mathbb{E}_s = [E_1, \cdots, E_{M+1}] \\
\mathbb{E}_n = [E_{M+2}, \cdots, E_N]
\] (4.4)

Therefore the complex \( N \)-dimensional observation space \( \mathcal{H} \) of the data covariance matrix \( \mathbb{R}_{xx} \) can be decomposed into two subspaces: the signal subspace \( \mathcal{H}_s \), with \( \text{dim}[\mathcal{H}_s] = M + 1 \), and the noise subspace \( \mathcal{H}_n \) with \( \text{dim}[\mathcal{H}_n] = N - M - 1 \), as shown in Figure 4.1. The signal subspace \( \mathcal{H}_s \) is spanned by either the manifold vectors \([S_d, S_1, S_2, \ldots, S_M]\) or the \( M + 1 \) dominant eigenvectors \([E_1, E_2, \ldots, E_{M+1}]\), and the noise subspace \( \mathcal{H}_n \) by the eigenvectors \([E_{M+2}, \ldots, E_N]\). Also, it is demonstrated in Figure 4.1 that the interference subspace \( \mathcal{L}[S_1, S_2, \ldots, S_M] \) is not equal to the subspace \( \mathcal{L}[E_1, E_2, \ldots, E_M] \) in the presence of the desired signal.
4.1 Introduction and Literature Review

4.1.1 Wiener-Hopf and Capon Beamformers

The Wiener-Hopf processor aims to maximize the SNIR at the array output, with the weight vector given by [4, 50]

$$\mathbf{w}_{\text{WH}} = \beta \mathbf{R}_{xx}^{-1} \mathbf{S}_d$$ (4.5)
where the scalar $\beta$ is independent of array output SNIR.

In [51], the well-known Capon beamformer has been presented to solve the following linearly constrained quadratic optimum problem:

$$
\min_w \quad w^H R_{xx} w \quad \text{subject to } \quad w^H S_d = 1 \quad (4.6)
$$

The above optimum problem aims to minimize the total output power while keeping the unity gain at the direction of the desired signal. Therefore the Capon beamformer is also called as the minimum-variance-distionless-response (MVDR) beamformer and its solution is given by

$$
w_{\text{Capon}} = \frac{1}{S_d R_{xx} S_d} R_{xx}^{-1} S_d \quad (4.7)
$$

Clearly, the Capon beamformer is a specific case of the Wiener-Hopf processor where $\beta$ in (4.5) becomes $\frac{1}{S_d R_{xx} S_d}$.

By defining the pointing vector $U_s = \beta S_d$ and substituting $R_{xx}$ given by Eq.(4.2) back into Eq.(4.5), the Wiener-Hopf weight vector may be rewritten as

$$
w_{\text{WH}} = E_s (D_s + \sigma^2 I_{M+1})^{-1} E_s U_s + \underbrace{\sigma^{-2} E_s U_s}_{=0} \quad (4.8)
$$

Then, with $y(t)$ denoting the output of the beamformer, the output power can be expressed as:

$$
P_{\text{out}} = E \left\{ y^2(t) \right\} = E \left\{ |w_{\text{WH}}^H \mathcal{E}(t)|^2 \right\} = \sigma_d^2 |w_{\text{WH}}^H S_d|^2 + \sum_{i=1}^{M} \sigma_i^2 |w_{\text{WH}}^H S_i|^2 + \sigma_n^2 |w_{\text{WH}}|^2 = P_{dout} + \sum_{i=1}^{M} P_{iout} + P_{nout} \quad (4.9)
$$
In the absence of any interference signals, \( P_{I_{\text{out}}} = 0 \) and both \( \mathbb{D}_s \) and \( \mathbb{I}_{M+1} \) become scalars and equal to \( N\sigma^2_d \) and 1 respectively. That is,

\[
\begin{align*}
\begin{cases}
P_{d_{\text{out}}} &= \sigma_d^2 |\mathbb{U}_s^H \mathbb{E}_s (\mathbb{D}_s + \sigma_d^2 \mathbb{I}_{M+1})^{-1} \mathbb{E}_s^H \mathbb{S}_d|^2 \\
P_{I_{\text{out}}} &= \sum_{i=1}^{M} \sigma_i^2 |\mathbb{U}_s^H \mathbb{E}_s (\mathbb{D}_s + \sigma_d^2 \mathbb{I}_{M+1})^{-1} \mathbb{E}_s^H \mathbb{S}_i|^2 \\
P_{n_{\text{out}}} &= \sigma_n^2 |\mathbb{U}_s^H \mathbb{E}_s (\mathbb{D}_s + \sigma_d^2 \mathbb{I}_{M+1})^{-2} \mathbb{E}_s^H \mathbb{U}_s + \frac{\sigma_n^2 \mathbb{U}_s^H \mathbb{P}_n \mathbb{U}_s}{\mathbb{U}_s^H \mathbb{P}_n \mathbb{U}_s} = 0
\end{cases}
\end{align*}
\tag{4.10}
\]

where the fact that \( \mathbb{P}_n \mathbb{S}_d = \mathbb{S}_d \) is used. From Eq.(4.11) it is clear that desired output power \( P_{d_{\text{out}}} \) is degraded with increasing desired signal power \( \sigma_d^2 \) at the input of the array, which is referred to as the *power inversion problem* [49]. For the desired signal with a high power at the array input, the corresponding desired signal power at the array output may suffer a sharp reduction which makes it difficult to detect it.

### 4.1.2 Modified Wiener-Hopf

The “modified” Wiener-Hopf processor overcomes the power inversion problem by filtering the desired signal from the array data and forming a covariance matrix without the desired signal. The weight vector is given by [49]

\[
\mathbf{w}_{m-WH} = \mathbb{P}_{i+n}^{-1} \mathbf{U}_s
\tag{4.12}
\]
where $R_{i+n}$, defined in (4.1), is formed by removing the effect of the desired signal from $R_{xx}$. The desired-signal-absent covariance matrix $R_{i+n}$ may be decomposed as follows

$$R_{i+n} = \sum_{i=1}^{N} \lambda_i E_i E_i^H$$

$$= E_I (D_I + \sigma_n^2 I_M) E_I^H + \sigma_n^2 E_O E_O^H$$

(4.13)

where the subscript ($\cdot_I$) and ($\cdot_O$) stand for interference and orthogonal with interference signals. $D_I$ is a diagonal matrix with the diagonal element the $M$ largest eigenvalues of $R_{i+n}$ arranged in decreasing order. The matrices $E_I$ and $E_O$ collect the eigenvectors associated with the interference signals and the eigenvectors orthogonal with interferences respectively, i.e.,

$$E_I = \begin{bmatrix} E_1, \ldots, E_M \end{bmatrix}$$

$$E_O = \begin{bmatrix} E_{M+1}, \ldots, E_N \end{bmatrix}$$

(4.14)

The decomposition of the observation space $H$ is as shown in Figure 4.2 where the interference subspace $H_I$ is spanned by $E_I$ or equivalently by $[S_1, \ldots, S_M]$, and the orthogonal subspace $H_O$ by $E_O$. 
Following the same analysis as with Wiener-Hopf, the optimum weight will be given by a similar equation to that of Eq.(4.8) but now the term \( \sigma_n^{-2} P_{E_O} U_s \) is not zero, i.e., \( \sigma_n^{-2} P_{E_O} U_s \neq 0 \). Then it can be shown that, in the absence of any interferences and taking into accounting that \( E_O \) spans the whole observation space, the projection operator applied to any vector belonging to that space leaves it unchanged. That is

\[
P_{E_O} S_d = S_d \quad \text{and} \quad P_{E_O} U_s = U_s
\]  

(4.15)
which implies that
\[
\begin{cases}
P_{\text{d, out}} = \sigma_d^2 \left( \frac{U^H S_d}{\sigma_n^2} \right)^2 \\
P_{\text{I, out}} = 0 \\
P_{\text{n, out}} = \frac{|U_s|^2}{\sigma_n^2}
\end{cases}
\tag{4.16}
\]

It is obvious from Eq.(4.16) that now the desired output at the output of the array is proportional to that at the input, so the power inversion problem is avoided.

When the pointing errors are absent ($U_s = S_d$) and the matrix inverse lemma [22] is used, one has
\[
W_{\text{WH}} = R_{xx}^{-1} S_d \\
= \left( R_{i+n} + \sigma_d^2 S_d S_d^H \right)^{-1} S_d \\
= \left( R_{i+n}^{-1} - \frac{R_{i+n}^{-1} \sigma_d^2 S_d S_d^H R_{i+n}^{-1}}{1 + \sigma_d^2 S_d S_d^H} \right) S_d \\
= \frac{1}{1 + \sigma_d^2 S_d S_d^H} R_{i+n}^{-1} S_d
\]
\tag{4.17}

The above shows that the weight vectors of the “full” and “modified” Wiener-Hopf processors differ only by a scalar factor and therefore offer identical maximum SNIR at the output if pointing errors are absent.

Now considering the presence of pointing errors, the output array SNIRs of both full and modified Wiener-Hopf processors can be expressed as [52]
\[
\text{SNIR}_{\text{WH, out}} = \frac{\sigma_d^2 S_d^H R_{i+n}^{-1} S_d \cos^2 \Psi}{1 + \left[ 2 \sigma_d^2 S_d^H R_{i+n}^{-1} S_d + (\sigma_d^2 S_d^H R_{i+n}^{-1} S_d)^2 \right] \sin^2 \Psi}
\]
\[
\text{SNIR}_{\text{m-WH, out}} = \sigma_d^2 S_d^H R_{i+n}^{-1} S_d \cos^2 \Psi
\]
\tag{4.18}
with

\[
\begin{align*}
\cos^2 \Psi & \triangleq \frac{\| U_s^H R_{i+n}^{-1} S_d \|^2}{\| U_s^H R_{i+n}^{-1} U_s \| \| S_d R_{i+n}^{-1} S_d \|} \\
\sin^2 \Psi & \triangleq 1 - \cos^2 \Psi
\end{align*}
\]  

(4.19)

where \( \Psi \) can be viewed as the generalized angle between the vector \( U_s \) and \( S_d \).

In the presence of pointing errors, \( \cos^2 \Psi < 1 \) and thus the output SNIRs of the both processors degrade. However, it can be seen clearly from (4.18) that the degradation of the modified Wiener-Hopf processor is much less than that of the full Wiener-Hopf processor if the term \( \sigma_d^2 S_d^H R_{i+n}^{-1} S_d \) is greater than unity. Thus, the modified Wiener-Hopf outperforms the full Wiener-Hopf in terms of the robustness to pointing errors.

However, both full and modified Wiener-Hopf always allow interference to pass at the output of an array of sensors, thus contaminating the desired signal. For this reason, the weight-vectors provided by these processors are not appropriate for complete interference cancellation.

### 4.1.3 Robust Techniques for Wiener-Hopf processors

In order to enhance the robustness of Wiener-Hopf processors (or Capon Beamformers), the popular diagonal loading (DL) method has been presented in [53] with the weight vector given by

\[
\begin{align*}
\overline{w}_{DL} &= \frac{(R_{xx} + \mu I)^{-1} U_s}{U_s^H (R_{xx} + \mu I)^{-1} U_s}
\end{align*}
\]  

(4.20)
where the pointing vector $U_s \neq S_d$ generally. Actually, the above is the solution for the following optimum problem

$$\min_w \ w^H R_{xx} w + \mu w^H w \quad \text{subject to} \quad w^H U_s = 1 \quad (4.21)$$

In comparison with standard Capon problem in (4.6), an additional penalty term $\mu w^H w$ is imposed on the objective function of the above DL problem. Thus the positive loading factor $\mu > 0$ can penalize large values of $w$ and detune the beamformer response around the nominal $U_s$ [45]. However, there is no systematic approach to choose the loading factor $\mu$. Usually, $\mu$ is set in an ad hoc way, typically $10\sigma_n^2$. Another drawback of the conventional DL is that there exists potential risk at over-penalizing the uncertainty of $R_{xx}$ (because of the quadratic term $w^H R_{xx} w$ in (4.21)) [54].

In recent years, the generalized versions of the DL, or called robust Capon beamformers (RCB), have been studied in [44, 45, 55–59], in which the diagonal loading factor is related to the uncertainty level and can be related to the uncertainty level. Among them, the RCB developed in [44] is the most influential, which can be stated by the following problem

$$\min_S \ S^H R_{xx}^{-1} S \quad \text{subject to} \quad \|S - U_s\|^2 \leq \epsilon \quad (4.22)$$

where the uncertainty level $\epsilon$ is a preselected constant satisfying

$$\|S_d - U_s\|^2 \leq \epsilon \quad (4.23)$$

The objective function in the above optimum problem is designed to minimize the output power of the Capon beamformer.

It is difficult to solve the above optimum problem directly because the
spherical constraint set is infinite. However, the optimal solution $S^\star$ must occur on the boundary of the constraint set, i.e., $\|S^\star - U_s\|^2 = \epsilon$. This can be understood by the following contradiction. Suppose that $\|S^\star - U_s\|^2 = \epsilon_1 < \epsilon$, then there exists a $\nu \in (0,1)$ such that

$$\|\nu S^\star - U_s\|^2 = \|(S^\star - U_s) + (\nu - 1) S^\star\|^2$$
$$\leq (\|S^\star - U_s\| + (1 - \nu) \|S^\star\|)^2$$
$$= (\sqrt{\epsilon_1} + (1 - \nu) \|S^\star\|)^2$$  \hspace{1cm} (4.24)

If $\nu$ is close to unity sufficiently, we can obtain $\|\nu S^\star - U_s\|^2 \leq \epsilon$, which means $\nu S^\star$ also belongs to the spherical constraint. Then,

$$(\nu S^\star)^H \mathbb{R}^{-1}_{xx} (\nu S^\star) = \nu^2 S^\star H \mathbb{R}^{-1}_{xx} S^\star < S^\star H \mathbb{R}^{-1}_{xx} S^\star$$  \hspace{1cm} (4.25)

The above means that one can find a vector in the constraint set, $\nu S^\star$, which makes the objective function in (4.22) less than that corresponding to $S^\star$. However, this is impossible because $S^\star$ is assumed to be the optimal solution and corresponds to the minimum of the objective function. Thus $S^\star$ has to be a boundary point of the spherical constraint. Therefore, the infinite number of inequality constraints in (4.22) is transformed into a single constraint (i.e., $\|S - U_s\|^2 = \epsilon$). Then the Lagrange multiplier methodology can be utilized to find the optimal solution. The associated weight vector may be expressed as

$$w_{RCB} = \frac{\mathbb{R}^{-1}_{xx} S^\star}{S^\star H \mathbb{R}^{-1}_{xx} S^\star}$$  \hspace{1cm} (4.26)

Interestingly, the RCB presented in [44] provides an equivalent solution to that in [57] and [45] where the convex optimization techniques are utilized.
It is evident that the solution $\mathbf{S}^*$ will no longer collinear to $\mathbf{S}_d$, even when the actual value of $\mathbf{S}_d$ is used in (4.22) instead of $\mathbf{U}_s$ [60]. Moreover, if a point is moved along the constraint boundary, the optimal solution $\mathbf{S}^*$ should be the point closest to the principal eigenvector of $\mathbf{R}_{xx}$. This implies that $\mathbf{S}^*$ is definitely affected by the interference signals because the principal eigenvector is a linear combination of all the manifold vectors (both the desired and the interferences). This situation would become worse when the DOAs of some interferences are close to the desired DOA, since the interference manifold vector may contribute significantly to the principal vector. In [60], a method based on multi-dimensional covariance fitting is presented to overcome the above problems. However, this method requires the knowledge of all the manifold vectors (including both the desired and the interference signals) and the associated uncertainty sets.

In [61–65] the method of signal-subspace projection (SSP) has been investigated for reducing the manifold mismatch, in which the presumed manifold is replaced with its projection onto the signal subspace. That is

$$\mathbf{w}_{SSP} = \mathbf{R}_{xx}^{-1}\mathbf{P}_E\mathbf{U}_s$$

(4.27)

Unfortunately this projection operation cannot eliminate the mismatch lying in the signal subspace. Also, both RCB and SSP are not an interference cancellation technique and hence the interferences can pass through the beamformers.

### 4.1.4 Blocking Matrix Approaches

Blocking matrix approaches [66–69] attempt to remove the effects of the desired signals by using a spatial blocking filter, as illustrated in Figure 4.3. The spatial
blocking matrix $\mathbb{B} \in \mathbb{C}^{N \times (N-q)}$ has the following properties

$$
\mathbb{B}^H \mathbb{S}_d = 0 \\
\mathbb{B}^H \mathbb{S}_i = c_i \mathbb{S}_i^{(q)} \quad i = 1, \ldots, M
$$

(4.28)

where $c_i$ is a complex constant, and $\mathbb{S}_i^{(q)}$ denotes the vector consisting of the first $(N-q)$ elements of $\mathbb{S}_i$. (4.28) means that the blocking matrix $\mathbb{B}$ is orthogonal to the manifold of the desired signal and hence the desired signal is blocked completely. In addition, the interferences can pass the spatial filter with dimensionality reduction by $q$. Let $\mathbb{E}_B \in \mathbb{C}^{(N-q)}$ denote the signal at the output of the spatial filter, which is given by

$$
\mathbb{E}_B = \mathbb{B}^H \mathbb{F}(t) \\
= \sum_{i=1}^{M} c_i m_i(t) \mathbb{S}_i^{(q)} + \mathbb{B}^H \mathbb{u}(t)
$$

(4.29)

Thus the interference subspace can be found by computing the $M$ dominant eigenvectors of the covariance matrix $\mathbb{R}_B = \mathbb{E}\{\mathbb{E}_B(t)\mathbb{E}_B^H(t)\}$. The weight vector $\mathbb{w}$ is obtained by projecting $\mathbb{S}_d$ onto the complement subspace of the interference subspace, which aims to cancel the interferences totally. Finally the received signal at the first $N-q$ sensors, i.e., $\mathbb{E}^{(q)}(t) = [x_1(t), x_2(t), \ldots, x_{N-q}(t)]^T$, is weighted by $\mathbb{w}$ to reproduce the desired signal.
$x(t) \in \mathbb{C}^N$

Form $\mathbb{B} \in \mathbb{C}^{N \times (N-q)}$ and perform eigen-decomposition

Construct weight vector by projecting the desired manifold onto the interference complement subspace

$y(t) \in \mathbb{C}^{(N-q)}$

$\mathbb{B} \in \mathbb{C}^{N \times (N-q)}$

Spatial blocking Filter

$x_{\hat{b}}(t) = \mathbb{B}^H x(t) \in \mathbb{C}^{(N-q)}$

Figure 4.3: The structure of the beamformer using blocking matrix
For the simple case where \( q = 1 \), the structure of the spatial blocking filter is depicted in Figure 4.4, where \( D \) is the distance between two adjacent sensors and the corresponding blocking matrix can be expressed as

\[
B = \begin{bmatrix}
-1, & 1, & 0, & \cdots, & 0, & 0 \\
0, & -1, & 1, & \cdots, & 0, & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0, & 0, & 0, & \cdots, & -1, & 1
\end{bmatrix}
\]

(4.30)

For the design details of \( B \) with general \( q \), see [67, 68]. Note that all the designs are derived strictly under the uniform linear array (ULA) assumption.

\[
x(t) \in C^N
\]

\[
\begin{align*}
x(t) &= e^{j\pi D \cos \theta_d} x_1(t) + e^{j\pi (N-2) D \cos \theta_d} x_3(t) + e^{j\pi (N-1) D \cos \theta_d} x_N(t) \\
x_B(t) &= B^H \left( S(\theta_d) \odot x(t) \right) \in C^{(N-1)}
\end{align*}
\]

Figure 4.4: An example of spatial blocking filter with \( q = 1 \)
By subtracting the received signals, the dimensionality of the observation space is reduced. For instance, the dimensionality is reduced by $q$ in Figure 4.3. In turn, this implies a reduction in resolution and $\text{SNIR}_{\text{out}}$. Furthermore, the approaches are strictly limited to ULA because the matrix $\mathbf{B}$ is derived on the basis of ULAs. In the presence of pointing errors, the desired signal will leak into the output of the blocking matrix, which may result in the severe performance degradation since the desired signal would be contained in the interference subspace and therefore be suppressed instead of being enhanced.

### 4.1.5 Problem Statement

On the basis of the above discussion, it is clear that the conventional Wiener-Hopf (or the Capon) beamformers suffer from the power inversion problem and lack of the robustness to the pointing errors. The existing robust Capon beamformers cannot work efficiently in many cases and allow the interferences to pass as well. Though the modified Wiener-Hopf overcome the power inversion problem and has the robustness to the pointing errors, the optimum weights for maximization of SNIR always allows interferences to pass to the output of the array. The blocking matrix methods achieve the complete cancellation of the interferences, but it cannot be applied in arrays with non-ULA geometries and it has the potential to cancel the desired signal in the presence of pointing errors. The problem to be addressed, in this chapter, is to find a unified solution which not only prevents power inversion and reduces the effect of pointing error but also provides, asymptotically, complete interference rejection that is lack in conventional beamformers.
4.2 Proposed Interference Cancellation Beamformer

In the previous chapter, it is demonstrated that in the presence of the pointing errors the actual manifold of the desired signal can be expressed as

\[ S_d = \mathbb{H} \hat{b} \]  

(4.31)

where the \( p \times 1 \) linear combination vector \( \hat{b} \) is unknown because \( S_d \) is not exactly known due to the pointing errors. The matrix \( \mathbb{H} \) is known or can be found easily, which is formed using one of the following expressions in this chapter

\[
\begin{align*}
\mathbb{H}_1 & = [S_0, \hat{S}(\theta_0), \tilde{S}(\theta_0)] \\
\mathbb{H}_2 & = [\hat{S}(\theta_0), \tilde{S}(\theta_0 - \Delta \theta), S(\theta_0 + \Delta \theta)]
\end{align*}
\]

(4.32)

where \( \hat{S}(\theta_0) \) and \( \tilde{S}(\theta_0) \) denote the first and second derivatives with respect to the presumed DOA \( \theta_0 \). \( \Delta \theta \) is related to the expected range of the DOA of the desired signal and can be set even if the actual DOA is unavailable.

It also assumes that \( [\mathbb{H} \ S_0, \ldots, S_M] \) is full column rank, which implies that the columns of \( [\mathbb{H} \ S_1, \ldots, S_M] \) are linearly independent. Moreover, the rank of \( [\mathbb{H} \ S_1, \ldots, S_M] \) is less than \( N \). The pointing vector is not equal to the actual manifold, i.e., \( U_s = S(\theta_0) \neq S_d \) (or written as \( U_s = S_0 \neq S_d \)).

Also, it is well known that the manifold vectors of all signals, including the desired signal and the interference signals, should belong to the so-called signal subspace \( \mathcal{L}[\mathbb{E}_s] \) which can be obtained by performing eigen-decomposition on the covariance matrix \( \mathbb{R}_{xx} \) (see Eq.(4.4)).

Now two different subspaces associated with the desired signal are at hand.
and thus it is intended to make the most of these information to remove the effects of the pointing errors as much as possible.

### 4.2.1 Estimate Using Vector Subspace Projections (VSP)

The approach proposed to eliminate the pointing errors applies the theorem of sequential VSP to find the intersection of the two constraint subspace. The simplest form of VSP, suggested by von Neumann in [70], is the alternating projection in which the iterative projections are performed to find the intersection of two Hilbert subspace. This work has been generalized for more than two closed subspaces and also for convex sets by Stark and Yang [71] so that the VSP method has found applications to a wide range of practical engineering problems. The applications of the projection methods to array signal processing have been considered in [72–75]. A fundamental theorem of VSP states the following:

**Theorem 4.1 [71]:** Let $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_m$ represent $m$ closed convex sets in a Hilbert space $\mathcal{H}$, and let $\mathbb{P}_{\mathcal{C}_i}$ denote the projection onto $\mathcal{C}_i$, $i = 1, \ldots, m$. If the intersection $\mathcal{C}_0 \triangleq \bigcap_{i=1}^{m} \mathcal{C}_i$ is non-empty and the dimension of $\mathcal{H}$ is finite, then the sequence $(\mathbb{P}_{\mathcal{C}_1} \mathbb{P}_{\mathcal{C}_2} \ldots \mathbb{P}_{\mathcal{C}_m})^k \mathbf{a}_0$ converges strongly to a point in the solution set $\mathcal{C}_0$, $\mathbb{P}_{\mathcal{C}_0} \mathbf{a}_0$, for every non-zero point $\mathbf{a}_0 \in \mathcal{H}$.

**Proof:** [71]

To place this theorem in the context of the pointing error elimination, the key is how to define the appropriate constraint sets $\{\mathcal{C}_i\}$ to describe the available information. Under the above assumptions, two constraints can now be imposed on $\mathbf{S}_d$.

\[
\mathcal{C}_1 = \{ \mathbf{S} : \mathbf{S} \in \mathcal{L}[\mathbb{E}_s] \} \quad (4.33a) \\
\mathcal{C}_2 = \{ \mathbf{S} : \mathbf{S} \in \mathcal{L}[\mathbb{H}] \} \quad (4.33b)
\]
The unknown manifold \( S_d \) is treated as a vector (or a point) lying in the intersection of \( C_0 \triangleq C_1 \cap C_2 \). Following Theorem 4.1, with the start point \( S_0 \) the sequence \( \{a_k\} \) generated by \[
 a_{k+1} = \mathbb{P}_{C_2} \mathbb{P}_{C_1} a_k \tag{4.34} 
\]
converges to \( S_d \) as \( k \to \infty \). The projection operations are defined as \[
 \mathbb{P}_{C_1} = \mathbb{E}_s \mathbb{E}_s^H \\
 \mathbb{P}_{C_2} = \mathbb{H} (\mathbb{H}^H \mathbb{H})^{-1} \mathbb{H}^H \tag{4.35} 
\]
This method is schematically depicted in Figure 4.5, where it is shown that the nominal manifold vector (i.e. point-A in Figure 4.5) is expected to converge to the actual manifold (point-B) after some iterations.

Here a solution using only one step is derived to avoid the iterative alternating projection. Consider what happens as \( k \to \infty \). Eq.(4.34) becomes \[
 a_\infty = \mathbb{P}_{C_2} \mathbb{P}_{C_1} a_\infty \tag{4.36} 
\]
The above equation implies that the final converged estimate $a_{\infty}$ is given by the eigenvector (up to a scaling factor) of matrix $P_{C_2}P_{C_1}$ which has a corresponding eigenvalue equal to one. Also, the maximum eigenvalue of the matrix product $P_{C_2}P_{C_1}$ is one. In accordance with Corollary 11 of [76], one obtains

$$eig_{\text{max}}(P_{C_2}P_{C_1}) \leq eig_{\text{max}}(P_{C_2}) \max_{u_{\text{H}}u=1} u^H P_{C_1} u$$

$$= \max_{u_{\text{H}}u=1} u^H P_{C_1} u$$

$$= eig_{\text{max}}(P_{C_1})$$

$$= 1 \quad (4.37)$$

where the fact that the maximum eigenvalues of the $P_{C_1}$ and $P_{C_2}$ are one is used in the above. The notation $eig_{\text{max}}(A)$ denotes the maximum eigenvalue of $A$. Hence if the nominal $\theta_0$ and $E_s$ are given, instead of $S_0$, the steering vector used is given by

$$\hat{U}_s = \beta P\{P_{C_2}P_{C_1}\} \quad (4.38)$$

where $\beta$ is chosen such that $\|\hat{U}_s\|^2 = N$.

### 4.2.2 Power Estimator

In [77], the power of the desired signal is estimated by searching the minimum of a cost function. The basic idea is that a temporary matrix $R(\alpha)$ is formed by removing the noise effect and subtracting the effect of the desired signal with a real variable factor $\alpha$ from $R_{xx}$, i.e.,

$$R(\alpha) = R_{xx} - \sigma_n^2 I - \alpha S_d S_d^H \quad (4.39)$$
Then the estimated power is the largest possible value of $\alpha$, which maintains $\mathbb{R}(\alpha)$ semipositive (for details, see Appendix 4.A). However, searching the minimum of the cost function may involve expensive computations because a number of eigen-decomposition operations are required.

In [55], the power estimation is solved by using covariance fitting theory, which can be expressed as follows

$$\hat{\sigma}_d^2 = \frac{1}{\Sigma_d^H \mathbb{R}_{xx}^{-1} \Sigma_d}$$ \hspace{1cm} (4.40)

Clearly the computation complexity is quite low because only one step is needed. Nevertheless the effect of noise on the signal subspace is neglected. This can be seen in the case where the interferences are absent. The covariance matrix becomes $\mathbb{R}_{xx} = \sigma_d^2 \Sigma_d S_d S_d^H + \sigma_n^2 I_N$ and its inverse is readily obtained

$$\mathbb{R}_{xx}^{-1} = \frac{I_N}{\sigma_n^2} - \frac{\sigma_d^2}{\sigma_n^2(\sigma_n^2 + N \sigma_d^2)} \Sigma_d S_d^H$$ \hspace{1cm} (4.41)

Then the power estimation using (4.40) is given by

$$\hat{\sigma}_d^2 = \sigma_d^2 + \frac{\sigma_n^2}{N} \neq \sigma_d^2$$ \hspace{1cm} (4.42)

Here a single step power estimation is proposed in which the effect of noise on the signal subspace is taken into account. The power of the desired signal $\sigma_d^2$
is equivalent to the largest possible value of $\sigma^2$ which satisfies

\[
eig_i (\mathbb{R}_{xx} - \sigma^2 S_d S_d^H) \geq \sigma^2_{n_i}, \quad \forall i
\]

\[\Leftrightarrow \quad \eig_i (\mathbb{R}_{xx} - \sigma^2_n \mathbb{I}_N - \sigma^2 S_d S_d^H) \geq 0, \quad \forall i
\]

\[\Leftrightarrow \quad \eig_i (\mathbb{E}_s^H (\mathbb{R}_{xx} - \sigma^2_n \mathbb{I}_N) \mathbb{E}_s - \sigma^2 \mathbb{E}_s^H S_d S_d^H \mathbb{E}_s) \geq 0, \quad \forall i
\]

\[\Leftrightarrow \quad \eig_i (\mathbb{D}_s - \sigma^2 \mathbb{E}_s^H S_d S_d^H \mathbb{E}_s) \geq 0, \quad \forall i
\]

\[\Leftrightarrow \quad \eig_i \left( \mathbb{I}_{M+1} - \sigma^2 \mathbb{D}_s^{-\frac{1}{2}} \mathbb{E}_s^H S_d S_d^H \mathbb{E}_s \mathbb{D}_s^{-\frac{1}{2}} \right) \geq 0, \quad \forall i
\]

\[\Leftrightarrow \quad 1 - \sigma^2 \mathbb{E}_s^H \mathbb{D}_s^{-1} \mathbb{E}_s \mathbb{E}_s^H S_d \geq 0
\]

\[\Leftrightarrow \quad \sigma^2 \leq \frac{1}{\mathbb{S}_d^H \mathbb{E}_s \mathbb{D}_s^{-1} \mathbb{E}_s^H S_d} \quad (4.43)
\]

where $\mathbb{D}_s^{-1/2}$ denotes Hermitian square root of $\mathbb{D}_s^{-1}$. The notation $\eig_i (A)$ represents the $i^{th}$ largest eigenvalue of $A$. In the above derivation, the fact that $\mathbb{D}_s^{-\frac{1}{2}} \mathbb{E}_s^H S_d S_d^H \mathbb{E}_s \mathbb{D}_s^{-\frac{1}{2}}$ is a rank-one matrix with the principal eigenvalue $\mathbb{S}_d^H \mathbb{E}_s \mathbb{D}_s^{-1} \mathbb{E}_s^H S_d$ is used. Hence the power of the desired signal is

\[
\sigma_d^2 = \frac{1}{\mathbb{S}_d^H \mathbb{E}_s \mathbb{D}_s^{-1} \mathbb{E}_s^H S_d} \quad (4.44)
\]

Let us examine the interference-absent case again. $\mathbb{D}_s$ becomes a scalar equal to $N\sigma_d^2$, and $\mathbb{S}_d^H \mathbb{E}_s \mathbb{E}_s^H S_d = N$. Thus (4.44) offers the precise power estimation.

The proposed method obtains the same power estimation as the approach in [77] but with much less computational complexity. Interestingly, [78] (see Lemma III.1 of [78]) proposed the equivalent power estimation method from the viewpoint of the oblique projection.
In the previous section, it has been seen that the modified Wiener-Hopf processor solves the power inversion problem through removing the desired signal effect from the covariance matrix. The proposed processor estimates and subtracts the desired signal effects to form the desired-signal-absent covariance matrix given by

\[ \hat{R_{i+n}} = R_{xx} - \sigma_d^2 \hat{U}_s (\hat{U}_s)^H \]  

(4.45)

where \( \hat{U}_s \) is obtained by Eq.(4.38) using the VSP method, the power \( \sigma_d^2 \) is computed by Eq.(4.44) using the modified one step power estimation method where \( \Sigma_d \) is replaced by \( \hat{U}_s \).

Performing eigen-decomposition on \( \hat{R}_{i+n} \) yields

\[ \hat{R}_{i+n} = \sum_{i=1}^{N} \lambda_i \hat{E}_i \hat{E}_i^H \]

\[ = \hat{R}_I (\bar{\Sigma}_I + \hat{\sigma}_n^2 I_M) \hat{E}_I^H + \hat{\sigma}_n^2 \bar{\Sigma}_O \hat{E}_O^H \]  

(4.46)

Now the interference subspace \( \mathcal{H}_I \) spanned by \( [\Sigma_1, \ldots, \Sigma_M] \) is equivalent to the subspace spanned by \( \hat{E}_I \). Let \( \mathbb{P}_I^\perp \) be the orthogonal projection operator of the interference subspace. Then \( \mathbb{P}_I^\perp \) can be estimated by

\[ \hat{\mathbb{P}}_I^\perp = I - \hat{E}_I \hat{E}_I^H \]  

(4.47)

The proposed interference cancellation weight vector is expressed as

\[ w_{ic} = \frac{\hat{\mathbb{P}}_I^\perp \hat{U}_s}{\sqrt{\hat{U}_s^H \hat{\mathbb{P}}_I^\perp \hat{U}_s}} \]  

(4.48)

The jammer power at the array output, using the actual DOA of the desired
4.2 Proposed Interference Cancellation Beamformer

signal, will be zero because $w_c$ is orthogonal with the interference subspace. The total output power of the array becomes

$$P_{\text{out}} = \sigma_d^2 \Sigma_d^H \Sigma_I^\perp \Sigma_d + \sigma_n^2$$

where the first term in the right is proportional to the input power of the desired signal. It is clear that the proposed method does not suffer from the power inversion problem. The SNR at the array output will be:

$$\text{SNR}_{\text{out}} = \frac{\sigma_d^2 \Sigma_d^H \Sigma_I^\perp \Sigma_d}{\sigma_n^2 \Sigma_d^H \Sigma_d}$$

The angle between the $\mathcal{H}_d$ (the subspace spanned the desired signal) and $\mathcal{H}_I^\perp$ subspaces, $\psi$, is defined as

$$\psi = \arccos \left( \frac{\Sigma_d^H \Sigma_I^\perp \Sigma_d}{\sqrt{\Sigma_d^H \Sigma_d \Sigma_d^H \Sigma_d}} \right)$$

Therefore, Eq.(4.50) is equivalent to:

$$\text{SNR}_{\text{out}} = \frac{\sigma_d^2}{\sigma_n^2} \left( \Sigma_d^H \Sigma_d \right) \cos^2 \psi = \frac{\sigma_d^2}{\sigma_n^2} N \cos^2 \psi$$

Eq.(4.52) shows that the ability of the proposed beamformer to completely cancel the interferences is obtained at the cost of SNIR degradation or partial cancellation of the desired signal. If there is an interference very close to the desired signal direction then $\psi \rightarrow 90^\circ$ and the output SNR deteriorates significantly, which may result in an unacceptable tradeoff. However, this undesirable property is not restricted to the proposed processor. The Wiener-Hopf processor (and all other known processes) also suffers from this restriction when interferences are located close to the desired signal direction.
4.2.4 Proposed Algorithm

The proposed algorithm can be presented as a series of steps as follows:

1. Estimate the data covariance matrix $R_{xx}$. Find its signal subspace eigenvectors $E_s$ and compute the projection operator $P_{C_1}$.

2. Calculate the $H$ using the nominal DOA $\theta_0$ and form the projection operator $P_{C_2}$.

3. Estimate the steering vector $U_s$ using Eq.(4.38) and the power $\sigma^2_d$ using Eq.(4.44) with $S_d$ replaced by $\hat{U}_s$.

4. Estimate the matrix $R_{i+n}$ by Eq.(4.45).

5. Perform eigen-decomposition on the matrix $R_{i+n}$. Find the projection operator $P_f^j$ using Eq.(4.47).

6. Compute the weight vector $w_{ic}$ using Eq.(4.48).

7. Weight the inputs of the array with $w_{ic}$.

4.3 Performance Analysis in the Presence of Pointing Errors

Now consider the case $\hat{U}_s \neq S_d$, i.e., there is a mismatch between the steering vector and the real manifold vector of the desired signal. Then the desired-signal-absent covariance matrix contains some perturbations as depicted in Figure 4.6,
4.3 Performance Analysis in the Presence of Pointing Errors

i.e.,

\[
\hat{R}_{i+n} = R_{xx} - \sigma_d^2 \hat{U}_s (\hat{U}_s)^H = R_{i+n} + \left( \sigma_d^2 S_d S_d^H - \sigma_d^2 \hat{U}_s (\hat{U}_s)^H \right) + \Delta \hat{R}_{i+n}
\]  \hspace{1cm} (4.53)

where the perturbation term \( \Delta \hat{R}_{i+n} \) is a rank-2 matrix if \( \hat{U}_s \neq S_d \).

Figure 4.6: Decomposition of the space by the columns of \( \hat{R}_{i+n} \)

Using the first-order subspace perturbation theory [79], the perturbed signal eigenvectors is given by

\[
\hat{E}_I = E_I + E_O E_O^H \hat{R}_{i+n} E_I D_I^{-1} + \Delta \hat{E}_I
\]  \hspace{1cm} (4.54)

where \( E_O \) and \( E_I \) are the noise-subspace eigenvectors and signal-subspace eigenvectors of the actual desired-signal-absent covariance matrix \( R_{i+n} \). Then the
perturbed orthogonal projection operator can be expressed by

\[
\begin{align*}
\hat{\mathbf{P}}_I^\perp & = \mathbb{I} - \hat{\mathbf{E}}_I \hat{\mathbf{E}}_I^H \\
& = \mathbf{P}_I^\perp + \mathbf{E}_I \hat{\mathbf{E}}_I^H + \mathbf{E}_I \hat{\mathbf{E}}_I^H + \mathbf{E}_I \hat{\mathbf{E}}_I^H \\
& = \mathbf{I} - \hat{\mathbf{E}}_I \hat{\mathbf{E}}_I^H
\end{align*}
\]

Substituting the above into Eq.(4.48) yields

\[
\hat{\mathbf{w}}_{ic} = \frac{\hat{\mathbf{P}}_I^\perp \hat{\mathbf{U}}_s}{\left| \hat{\mathbf{P}}_I^\perp \hat{\mathbf{U}}_s \right|^2}
\]

Now the output interference power, \( P_{I_{out}} \), no longer equals zero due to pointing errors, because the weight vector is not orthogonal with the interference manifold, i.e., \( \hat{\mathbf{w}}_{ic}^H S_i \neq 0 \).

Consequently \( \text{SNIR}_{out} \) in the presence of pointing errors can be given by

\[
\text{SNIR}_{out} = \frac{\sigma_d^2 \left| \hat{\mathbf{w}}_{ic}^H S_d \right|^2}{P_{I_{out}} + \sigma_n^2 \left| \hat{\mathbf{w}}_{ic}^H \hat{\mathbf{w}}_{ic} \right|^2} = \frac{\sigma_d^2 \left| \hat{\mathbf{U}}_s^H \hat{\mathbf{P}}_I^\perp S_d \right|^2}{\sum_{i=1}^{M} \sigma_i^2 \left| \hat{\mathbf{U}}_s^H \hat{\mathbf{P}}_I^\perp S_i \right|^2 + \sigma_n^2 \left| \hat{\mathbf{U}}_s^H \hat{\mathbf{P}}_I^\perp \hat{\mathbf{U}}_s \right|^2} = \frac{\sigma_d^2 N \cos^2 \hat{\psi}}{\sum_{i=1}^{M} \sigma_i^2 \left| \hat{\mathbf{U}}_s^H \hat{\mathbf{P}}_I^\perp S_i \right|^2} + \sigma_n^2
\]

where \( \hat{\psi} \) denotes the angle between \( S_d \) and \( \hat{\mathbf{P}}_I^\perp \hat{\mathbf{U}}_s \). When the pointing errors are absent, the perturbation term \( \hat{\mathbf{P}}_I^\perp \) is a matrix of zeros and \( \hat{\psi} \) equals to \( \psi \), which means the above equation reduces to Eq.(4.52).
4.4 Simulation Studies

Assume that one desired signal and three interferences impinge on a ULA with $N = 10$ isotropic sensors and half-wavelength sensor spacing. Note that the proposed algorithms is applicable to arbitrary array geometry although here ULA array is used. The desired signal’s DOA is fixed at $90^\circ$ in all the following examples. All signals have unit power. Noise power is $\sigma_n^2 = 0.1$ (-10dB). The matrix $\mathbf{H}$ is composed of three columns (the nominal manifold vector and its first two derivatives with respect to the DOA. See Eq.(3.6)).

**Case 1:** In the first example, the DOAs of interference are at $60^\circ$, $62^\circ$ and $70^\circ$. The pointing angle is $90^\circ$ (i.e., no pointing error in this example). The array patterns of three methods (Wiener-Hopf, Lee-Lee [68] and the proposed) are shown in Figure 4.7, where Lee-Lee’s method is one of the blocking matrix methods. It can be seen in Figure 4.7 that by using the Wiener-Hopf processor it is not possible to distinguish (resolve) the two closely spaced interfering sources ($60^\circ$ and $62^\circ$) correctly whereas it is possible with the processor based on the proposed $\mathbf{w}_c$ or on Lee-Lee’s weight. Also, both Lee-Lee’s method and the proposed method present very deep nulls at the locations of the unknown interferences while they provide a free way to the desired signal. An added advantage is that each deep null provides the location estimate of an unknown interference, because these deep nulls are easily distinguishable from the rest of the array pattern. It can also be seen in Figure 4.7 that Wiener-Hopf allows the interference to pass through partially because the associated array pattern cannot place deep nulls at the locations of interferences. The power estimations of the three methods are listed in Table 4.1 which illustrates that the proposed method, with much less computational complexity, provides the precise estimate as well as [77], whereas there exists the estimation error in [55].
Figure 4.7: Array pattern with two interferences close together (60° and 62°). The DOA of the desired signal is 90°.

Table 4.1: Estimated power of the desired signal

<table>
<thead>
<tr>
<th></th>
<th>actual power</th>
<th>Proposed</th>
<th>Ref. [77]</th>
<th>Ref. [55]</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1.0191</td>
</tr>
</tbody>
</table>

Case 2: Comparison is made with respect to pointing error effects in the Wiener-Hopf ("modified" and "full" version), proposed processor and Lee-Lee processor in the second example. The DOAs of interferences are 70°, 80° and 100°. Though Figure 4.8 shows that the proposed processor is more susceptible to pointing errors than the "modified" Wiener-Hopf solution, it performs much better than the Lee-Lee and "full" Wiener-Hopf processors. The SNIR of Lee-Lee without
pointing error is below the optimum SNIR. This is because the dimensionality of
the observation space is reduced in the Lee-Lee processor to construct the blocking
matrix. In addition, Table 4.2 shows that the results computed by the analytical
equations in previous section are consistent with the simulation results when the
pointing errors are small. If the pointing error is over 4°, the approximation
in Eq.(4.54) in the sense of the first-order subspace perturbation is invalid and
therefore there are significant mismatches between the simulation results and the
theoretical analysis.

![Graph](image)

Figure 4.8: Effects of pointing errors with noise level at -10dB. The true
direction of the desired signal maintained at 90°.
Table 4.2: SNIR\textsubscript{out} (unit: dB) with noise level at -10dB

<table>
<thead>
<tr>
<th>pointing angle</th>
<th>SNIR\textsubscript{out} by simulation</th>
<th>SNIR\textsubscript{out} by Eq. (4.57)</th>
</tr>
</thead>
<tbody>
<tr>
<td>85(^0)</td>
<td>0</td>
<td>8.12</td>
</tr>
<tr>
<td>86(^0)</td>
<td>9.64</td>
<td>10.10</td>
</tr>
<tr>
<td>87(^0)</td>
<td>17.44</td>
<td>16.86</td>
</tr>
<tr>
<td>88(^0)</td>
<td>19.46</td>
<td>19.41</td>
</tr>
<tr>
<td>89(^0)</td>
<td>19.55</td>
<td>19.55</td>
</tr>
<tr>
<td>90(^0)</td>
<td>19.55</td>
<td>19.55</td>
</tr>
<tr>
<td>91(^0)</td>
<td>19.55</td>
<td>19.55</td>
</tr>
<tr>
<td>92(^0)</td>
<td>19.51</td>
<td>19.49</td>
</tr>
<tr>
<td>93(^0)</td>
<td>18.89</td>
<td>18.59</td>
</tr>
<tr>
<td>94(^0)</td>
<td>14.68</td>
<td>13.90</td>
</tr>
<tr>
<td>95(^0)</td>
<td>5.86</td>
<td>7.74</td>
</tr>
</tbody>
</table>

Case 3: In the third example, three other existing robust beamformers are also simulated in the same environment as the second example. The first robust algorithm is the conventional DL [53] with the diagonal loading factor \(\xi = 10\sigma_n^2\) (see Eq.(4.20)). The second method is the SSP method proposed in [61] and its weight vector is given by

\[
w_{\text{ssp}} = R_{xx}^{-1}P_c S(\theta_0)\tag{4.58}
\]

Two different RCBs (with the spherical constraint and with flat ellipsoidal constraint) presented in [44] are examined. The legend RCB-1 in Fig.4.9 stands for the RCB with the spherical constraint. Here the spherical boundary \(\epsilon\) \((||S(\theta_d) - S(\theta_0)||^2 \leq \epsilon\) is equal to the larger of \(||S(\theta_0) - S(\theta_0 - \Delta\theta)||^2\) and \(||S(\theta_0) - S(\theta_0 + \Delta\theta)||^2\), where \(\Delta\theta = |\theta_0 - \theta_d| + 0.5^\circ\). The legend RCB-2 in Fig.4.9 means the RCB with flat ellipsoidal constrain. In previous chapter it has been shown that the flat ellipsoidal uncertainty set in [44] can be viewed as an ex-
Simulation Studies

Sample of the uncertainty model used in this thesis. Setting \( \mathbb{B} = [\dot{S}(\theta_0), \ddot{S}(\theta_0)] \), the condition that \( S^H(\theta_0)(\mathbb{B}^\dagger)^H \mathbb{B}^\dagger S(\theta_0) > 1 \) (where \( \dagger \) denotes the Moore-Penrose pseudo-inverse) does not hold. This means that RCB with \( \mathbb{B} = [\dot{S}(\theta_0), \ddot{S}(\theta_0)] \) fails (see (45) of [44]). In order to satisfy this condition and simulate the second RCB method, \( \mathbb{B} \) is set to be \( \mathbb{B} = [S(\theta_0), S(\theta_0) - S(\theta_0 - \Delta \theta), S(\theta_0) - S(\theta_0 + \Delta \theta)] \) The legends Proposed-1 and Proposed-2 in Figure 4.9 represent the proposed method using \( \mathbb{H}_1 = [S(\theta_0), \dot{S}(\theta_0), \ddot{S}(\theta_0)] \) and \( \mathbb{H}_2 = [S(\theta_0), S(\theta_0) - S(\theta_0 - \Delta \theta), S(\theta_0) - S(\theta_0 + \Delta \theta)] \) respectively. Figure 4.9 shows that the proposed method outperforms the other three methods when pointing errors happen. In addition, the proposed method is applicable to more practical applications than the RCB with flat ellipsoidal constraint because the condition \( S^H(\theta_0)(\mathbb{B}^\dagger)^H \mathbb{B}^\dagger S(\theta_0) > 1 \) is relaxed.

![Figure 4.9: Effects of pointing errors. The true direction of the desired signal maintained at 90°.](image)

Case 4: Finally, the effect of finite snapshots is tested. The simulation environ-
ment is still the same as the second example. The pointing angle is always 88°, i.e., 2° pointing error. The array output SNIRs of four methods versus the snapshot number varying from 20 to 2000 are plotted in Figure 4.10. This example also indicates that the proposed method is superior to others.

![Figure 4.10: The effect of finite snapshots. The true direction of the desired signal maintained at 90°. The pointing angle is 88°.](image)

### 4.5 Summary

The actual manifold vector of the desired signal differs from the nominal one when pointing errors occur. However, the actual manifold can be assumed to lie in a linear subspace, for instance, which is spanned by the nominal manifold and the derivative vectors around the nominal DOA. Moreover, the true manifold vector belongs to the signal subspace which can be found by performing eigen-decomposition on the covariance matrix of the received array data. Then the
estimate of the desired signal manifold can be obtained using VSP, by means of which much of the pointing error is eliminated. The power of the desired signal can be computed in one step operation using the theory of covariance fitting. The estimates of manifold and power of the desired signal are then utilized to construct the proposed estimate-and-subtract interference canceller beamformer in which the power inversion problem is avoided. In the absence of pointing errors, the proposed beamformer can completely cancel the interferences at the cost of SNIR degradation or partial cancellation of the desired signal.

The tasks accomplished by the proposed algorithm are summarized as follows

- provides complete interference cancellation with the output of the array consisting only of desired signal and thermal noise;
- provides reduced susceptibility to pointing errors and noise level;
- does not suffer from the power inversion problem.
4.6 Appendix 4.A Revisit the power estimation

In this section, the power estimator proposed in [77] is revisited. A cost function $\xi(\alpha)$ is defined in [77] as

$$
\xi(\alpha) = \sum_{i=1}^{N} (1 + \text{eig}_i(\mathbb{R}(\alpha))) + 10 \log_{10} \left( \sum_{i=1}^{N} |\text{eig}_i(\mathbb{R}(\alpha))| \right) \quad (4.59)
$$

where the matrix $\mathbb{R}(\alpha)$ is formed from $\mathbb{R}_{xx}$ by excising the effects of the noise and the desired signal, i.e.,

$$
\mathbb{R}(\alpha) = \mathbb{R}_{xx} - \sigma_d^2 I_N - \alpha \mathbb{S}_d \mathbb{S}_d^H \quad (4.60)
$$

[77] argues that the power can be found by searching the minima of $\xi(\alpha)$ over $\alpha$. That is

$$
\hat{\sigma}_d^2 = \arg \min_{\alpha} (\xi(\alpha)) \quad (4.61)
$$

However, [77] has not elaborated why this method can work. Next, the essence of this method will be revealed.

Substituting $\mathbb{R}_{xx}$ in (4.1) to (4.60) produces

$$
\mathbb{R}(\alpha) = \sum_{i=1}^{M} \sigma_i^2 \mathbb{S}_i \mathbb{S}_i^H + (\sigma_d^2 - \alpha) \mathbb{S}_d \mathbb{S}_d^H = \mathbb{R}_i + (\sigma_d^2 - \alpha) \mathbb{S}_d \mathbb{S}_d^H \quad (4.62)
$$

where the matrix $\mathbb{R}_i \triangleq \sum_{i=1}^{M} \sigma_i^2 \mathbb{S}_i \mathbb{S}_i^H$ has the eigenvalues

$$
eig_1(\mathbb{R}_i) \geq eig_2(\mathbb{R}_i) \geq \ldots \geq eig_M(\mathbb{R}_i) > eig_{M+1}(\mathbb{R}_i) = \ldots = eig_N(\mathbb{R}_i) = 0 \quad (4.63)
$$

This means that $\mathbb{R}_i$ has $M$ positive eigenvalues and $(N - M)$ null eigenvalues,
if the theoretical covariance matrix $R_{xx}$ is used. In the case of $\alpha \leq \sigma_d^2$, $R(\alpha)$ is a semipositive matrix and thus has no negative eigenvalues. Therefore, only the first term in the right side of (4.59) affects $\xi(\alpha)$, and thus $\xi(\alpha)$ may be simplified as

$$
\xi(\alpha) \leq \sigma_d^2 = \sum_{i=1}^{N} (1 + \text{eig}_i(R(\alpha)))
$$

$$
= (M + 1) + \text{tr}(R(\alpha))
$$

$$
= (M + 1) + N \sum_{i=1}^{M} \sigma_i^2 + N(\sigma_d^2 - \alpha)
$$

(4.64)

Obviously, $\xi(\alpha)$ monotonically decreases in the area $\alpha \in [0, \sigma_d^2]$.

Now consider the case of $\alpha > \sigma_d^2$. Then $\sigma_d^2 - \alpha < 0$. It is reasonable to assume that $R(\alpha)$ is of rank-$(M+1)$ because $[S_d, S_1, \ldots, S_M]$ is linearly independent. Hence $R(\alpha)$ has $(M+1)$ non-zero eigenvalues. According to the interlacing eigenvalue theorem 8.1.8 in [22], it is found that

$$
eig_N(R(\alpha)) \leq \cdots \leq \eig_{M+1}(R(\alpha)) \leq \eig_M(R_i) = 0 \leq \eig_M(R(\alpha)) \leq \cdots \leq \eig_1(R(\alpha))
$$

(4.65)

which implies that the smallest eigenvalue, $\eig_N(R(\alpha))$, must be a negative value and all the other non-zero eigenvalues of $R(\alpha)$ are positive. Then the cost function can be rewritten as

$$
\xi(\alpha > \sigma_d^2) = \sum_{i=1}^{M} (1 + \text{eig}_i(R(\alpha))) + 10 \log_{10} |\text{eig}_N(R(\alpha))|
$$

(4.66)

Suppose $\delta = \alpha - \sigma_d^2 > 0$ is a sufficiently small value, then the value of $|\text{eig}_N(R(\alpha))|$ is much less than unity. This implies that the term $10 \log_{10} |\text{eig}_N(R(\alpha))|$ would result in a sharp decrease. Then with the increase of $\alpha$, the eigenvalue $\text{eig}_N(R(\alpha))$ becomes more negative. However, the absolute
value $|\text{eig}_N(\mathbb{R}(\alpha))|$ becomes larger. In turn, the $10\log_{10}(\cdot)$ factor would cause a sharp increase, when $10\log_{10}|\text{eig}_N(\mathbb{R}(\alpha))|$ dominates $\xi(\alpha)$.

In sum, in the area $\alpha \in [0, \sigma^2_d)$, the cost function $\xi(\alpha)$ monotonically decreases and $\mathbb{R}(\alpha)$ has $(M + 1)$ positive eigenvalues. At the point $\alpha = \sigma^2_d$, $\xi(\alpha)$ reaches the first minima and $\mathbb{R}(\alpha)$ has $M$ positive eigenvalues only. When $\alpha > \sigma^2_d$, $\xi(\alpha)$ monotonically increases and $\mathbb{R}(\alpha)$ has $M$ positive eigenvalues and one negative eigenvalue.
Chapter 5

Conclusions

This thesis has been concerned with two fundamental problems in the array signal processing. The first problem is DOA estimation. For the arbitrary arrays, the extended root-MUSIC obtains the DOA information by rooting a polynomial and thus is categorized as a search-free DOA estimate methods. In Chapter 2, three major approaches of the extended root-MUSIC are revisited firstly. The problem that they face is the high computational complexity due to the high order of the polynomial. In order to reduce the computational burden, two different solutions are proposed. One is implemented in an iterative way and another can be realized by scanning multiple circles in parallel. Both proposed methods are insensible to the radial errors and therefore outperform the MUSIC-type methods.

Chapter 3 and 4 focus on the design of beamformers robust to pointing errors. The pointing errors make the presumed manifold different than the actual manifold. However, a linear subspace can be found by using the knowledge of the presumed manifold, which contains the actual manifold. A matched direction beamforming (MDB) is presented in Chapter 3, which is robust to
1. pointing errors,

2. overestimation of the signal subspace dimension.

However, this MDB beamformer cannot cancel the interference completely because it is of Wiener-Hopf type. In Chapter 4, a novel interference cancellation beamformer is proposed, which provides a unified solution to the problems of

1. power inversion,

2. the robustness to pointing errors,

3. complete interference cancellation.

Much of pointing errors is removed by using VSP projections. The power inversion problem is avoided by excising all the effects of the desired signal from the covariance matrix. The interference signals are completely cancelled at the array output using a weight vector which is orthogonal with the interference subspace.

5.1 List of Contributions

The following list summarizes the main contributions of this thesis which are supported by the publications given in page 7 of the thesis.

1. Proposal of a framework consisting of spectral factorization (using Schur algorithm) and large eigenvalue finding (using Arnoldi iteration) to reduce the computational complexity of the extended root-MUSIC. By exploiting the properties that the roots appear in conjugate reciprocal pairs and the roots of interest are closest to the unit circle, the proposed method computes the desired roots only, implying a large number of the unwanted roots are exempt from the calculation. (Chapter 2)
2. Proposal of an IDFT-base root-MUSIC in which the root magnitude is explicitly modelled and the polynomial obtained by the extended root-MUSIC is reformulated to a typical form of IDFT. The rooting process is avoided by the implementation of multiple IDFT operations in parallel. (Chapter 2)

3. Development of a Wiener-Hopf type MDB beamformer which is robust to pointing errors. The pointing errors result in the mismatch between the nominal direction and the actual direction of the desired signal. This provides a mismatch between the corresponding manifold vectors (nominal and true manifold vectors). However, a linear subspace containing the true manifold vector can be found by using the knowledge of the nominal manifold. Then the linear combination vector associated with true manifold is equivalent to the principal eigenvector of a certain rank-1 matrix up to scalar. Importantly, this rank-1 matrix is obtained without the recourse to the knowledge of interference subspace. Moreover, it is found that the proposed MDB is also robust to the overestimation of signal subspace dimension. (Chapter 3)

4. Development of a pointing error elimination method by using VSP projections. Two implementations are provided: the alternative iteration and a single step solution. (Chapter 4)

5. Derivation of a single step subspace-type power estimator using the covariance fitting theory in which the noise effect is taken into account. It enjoys both the high accuracy estimation and computational effectiveness. (Chapter 4)

6. Proposal of a unified solution to the problems of power inversion, robust-
ness to pointing errors, and complete interference cancellation. The desired-
signal-absent covariance matrix is formed by excising the effects of the de-
sired signal where the estimations of the power and manifold vector are
utilized. The proposed weight vector is found by projecting the manifold of
the desired signal onto the complement of the interference subspace (which
is spanned by the signal vectors of the desired-signal-absent covariance ma-
trix). Performance analysis of the proposed interference cancellation beam-
former in the presence of pointing errors is also presented, where the sub-
space perturbation theory is used. An analytic expression of output SNIR
is derived, which is verified by the simulation results. (Chapter 4)

5.2 Suggestions for Further Work

A number of different problems associated with DOA estimation for arbitrary
arrays and beamformers robust to pointing errors have been investigated in this
thesis, but the following topics are still opened for additional research effort.

1. Two dimension DOA estimation for arbitrary arrays using root-MUSIC.
Almost all extended root-MUSIC algorithms aim at finding the azimuth
angle only and leave the elevation angle untouched. By using the manifold
separation technique (MST), the array manifold, which is the function of
both azimuth and elevation angles, can be modelled using spherical har-
monics. Then the MUSIC spatial spectrum can be written in polynomial
form. Thus each peak in the MUSIC spectrum corresponds to one root of
this polynomial, meaning that the 2-dimension DOA angles can be found
by rooting the polynomial instead of searching minima over $(\theta, \phi)$ sphere.
However, the problem needed to address is how to find the roots in the
2-dimension complex space \((\exp(-j\theta), \exp(-j\phi))\), which can be viewed as a 4-dimension real space). An immediate consequence is that the roots are not isolated points but rather 2-dimension surfaces in the 4-dimension space \([19]\).

2. **Extension of IQML/MODE to arbitrary arrays.** The classical MUSIC and root-MUSIC are subspace-based DOA estimations which can achieve the performance as ML estimators (MLE) when the snapshot number is sufficiently large and the signal sources are uncorrelated. However, when the sources are coherent or the snapshot number is relatively small, the performance of subspace-type estimators will suffer a degradation in comparison with MLE. As stated in Chapter 2, a major drawback of ML methods is their computational intensity due to the requirement of multi-dimension search. In order to reduce the computational complexity, the iterative quadratic maximum likelihood (IQML) \([80, 81]\) and the method of direction estimation (MODE) \([82, 83]\) reformulate the multi-dimension search into a problem of polynomial rooting. These two MLEs have the computational complexity comparable to that of MUSIC but better performance in the case of highly correlated sources. However, these two MLEs are derived under the assumption of ULA arrays. Hence the extension of these MLEs to arbitrary arrays is attractive. In \([84]\), MODE has been extended to arbitrary arrays using the technique of interpolated array. The main problem of \([84]\) is that the significant mapping errors are introduced, which degrades the estimation performance. The MST technique cannot directly be used for the extension because the matrix \(G_r\) is not of full column rank but of full row rank. The future works should provide the similar estimation performance to the MLEs using ULA but with rather low computational
3. **Robust transceiver design.** The robust techniques developed in this thesis are applied at the receiver end. The transceiver may have a variety of structures. For instance, both the transmitter and receiver are equipped with an antenna array, which is called vector-in-vector-out (VIVO) system. Also, the feedback from the receiver to the transmitter may contain partial or full channel information. The robust transceiver may place the emphasis on how the cooperation among the transceiver is designed to combat different uncertainty types (such as pointing errors, moving sources, uncalibrated array) in different situations.

4. **Robust wideband array beamformer.** In wireless communications, the high data rate benefits from a wideband system. Using wideband signals in a radar system leads to higher resolution than the narrowband counterpart. However, the wideband applications violate the assumption that baseband signal amplitudes at different sensors are approximately identical, which renders the array manifold model for narrowband invalid. Generally the tapped-delay-line structure is used at the receiver or/and the transmitter. Thus the array system needs to handle in two domains (space-time processing). In such case, robust wideband array beamformer is required to overcome different uncertainty problems.
Bibliography


