The University Of Sheffield.

Beamforming and Direction of Arrival Estimation Based on Vector Sensor Arrays

By

Xiang Lan

A doctoral thesis submitted in fulfilment of the requirements for the award of

Doctor of Philosophy

The University of Sheffield

Department of Electrical and Electronic Engineering

March 2019

Supervisor

Dr. Wei Liu

Beamforming and Direction of Arrival Estimation Based on Vector Sensor Arrays

Xiang Lan

PHD THESIS

Communications Group

Department of Electrical and Electronic Engineering

The University of Sheffield

March 2019

Abstract

Array signal processing is a technique linked closely to radar and sonar systems. In communication, the antenna array in these systems is applied to cancel the interference, suppress the background noise and track the target sources based on signals'parameters. Most of existing work ignores the polarisation status of the impinging signals and is mainly focused on their direction parameters. To have a better performance in array processing, polarized signals can be considered in array signal processing and their property can be exploited by employing various electromagnetic vector sensor arrays.

In this thesis, firstly, a full quaternion-valued model for polarized array processing is proposed based on the Capon beamformer. This new beamformer uses crossed-dipole array and considers the desired signal as quaternion-valued. Two scenarios are dealt with, where the beamformer works at a normal environment without data model errors or with model errors under the worst-case constraint. After that, an algorithm to solve the joint DOA and polarisation estimation problem is proposed. The algorithm applies the rank reduction method to use two 2-D searches instead of a 4-D search to estimate the joint parameters. Moreover, an analysis is given to introduce the difference using crossed-dipole sensor array and tripole sensor array, which indicates that linear crossed-dipole sensor array has an ambiguity problem in the estimation work and the linear tripole sensor array avoid this problem effectively. At last, we study the problem of DOA estimation for a mixture of single signal transmission (SST) signals and duel signal transmission (DST) signals. Two solutions are proposed: the first is a two-step method to estimate the parameters of SST and DST signals separately; the second one is a unified one-step method to estimate SST and DST signals together, without treating them separately in the estimation process.

Contents

List of Publications

Journal paper

1. B. Zhang, W. Liu and X. Lan. "Orthogonally polarised dual-channel directional modulation based on crossed-dipole arrays." IEEE Access, DOI: 10.1109/Access. 2019.2903909, 2019.

2. X. Lan, and W. Liu. "Fully quaternion-valued adaptive beamforming based on crossed-dipole arrays." Electronics 6(2), 34, June, 2017.

Conference paper

1. X. Lan, and W. Liu. "Joint 4-D DOA and polarisation estimation based on linear tripole arrays." In Proc. 22nd International Conference on Digital Signal Processing (DSP), London, UK, August, 2017.

Under Review

1. X. Lan, and W. Liu. "Joint 4-D DOA and polarisation estimation with crossed-dipole and tripole sensor arrays." IEEE Transactions on Aerospace and Electronic Systems (TAES).

In Preparation

1. X. Lan, and W. Liu."Direction finding for signals with a mixed signal transmission model."

List of Figures

List of Tables

Acknowledgements

I would like to express my great gratitude and sincere appreciation to my supervisors Dr. Wei Liu and Dr. Jonathan M Rigelsford. Thank you for guiding me into the field and helping me so much with my research.

To my parents, thank you for giving me the chance to have my PhD study in Sheffield. Your support and encouragement will be the best present in my life forever.

To Huang Huang, Tang Ning and all friends, thank you for standing by my side all the time. The life becomes colourful with all of you.

To all colleagues in the group, thank you for the support these years. I will keep the working experience with you in my memory as this experience will be the most precious wealth in my life.

Chapter 1

Introduction

1.1 Research Background

Array signal processing techniques have been widely used in many applications. In modern radar and sonar, antenna and hydrophone array is an essential component in the system to detect and locate the targets. Many communication systems use microphones or loudspeakers or multiple beam antennas to achieve their performance objectives. Seismic arrays are designed for oil exploration, underground nuclear tests detection and even earthquake prediction [1]. Various medical diagnosis and treatment techniques exploit ultrasound arrays [2, 3]. Large antenna arrays can help to achieve high resolution goals in radio astronomy [4].

Specifically, array signal processing can be divided into three different sub-areas: signal detection, beamforming and direction of arrival (DOA) estimation. Detection theory or signal detection theory is a means to quantify the ability to distinguish between signals and noise with information of the received array signals [5]. Beamforming is a widely used technique in tracking and receiving signals from a selected direction or directions [6]. Sometimes we consider beamforming as spatial filtering [7]. It can be applied to both the transmitting and receiving ends to achieve spatial selectivity. This is realised by combining elements in a sensor array in such a way that signals at particular angles experience

constructive combination, while others are added together destructively. Generally, there are three different types of array structures for array signal processing: the linear array, where all the sensors are placed along a straight line; the planar array, where the sensors are spaced over a two-dimensional (2D) surface [8, 9]; the volumetric arrays, where the sensors are placed spatially in a three-dimensional (3D) space [10]. DOA estimation in array signal processing is used to estimate the direction of incoming signals by an array of sensors, which is widely used in radar detection, seismology, wireless communications and radio astronomy [11, 12]. In the following, some application areas are introduced.

Radar is probably the first application area of array processing. Although the primary concept of phased antenna array was known during World War I, the first application was during World War II. Most kinds of radar work in the defense sector. Examples can be given as the fire control radars in Navy ships, high resolution bombing radars and height-finding radars. Non-military radars are used for navigation, meteorologic, velocity measurement and air traffic control.

Array signal processing also plays a very important role in seismology. One important area is to detect and locate underground nuclear explosions. In the 1960s and 1970s many useful developments were achieved in this area. One of the most famous beamformer, Capon's minimum variance distortionless response (MVDR) beamformer was developed in that period [13], which will be discussed in the following chapter. Another topic related to seismology is of great significance to us, that is the exploration seismology [14], where the objective process is to construct an image of the subsurface in which the structure and physical properties are described.

In many modern communication systems, antenna arrays, or sensor arrays are widely used. Signals in a communication system are normally point sources. Due to the channel characteristics, we can consider the signals arriving at a sensor array as a single plane wave, multiple plane waves or as spatially spread signals [15]. In addition to receiver noise, the interference may include other communication signals or intentional jamming signals. Moreover, phased antenna array is an important component in satellite systems.

For example, the Defense Satellite Communication System (DSCS III) has a 61-beam receive MBA (multiple beam antennas) and 19-beam transmit MBA. Many of the low earth orbit (LEO) satellite systems utilize phased arrays.

1.2 Original Contributions

In our research, the sources are considered to emit polarized signals, and we employ vector sensor (crossed-dipole or tripole sensor) arrays to process the received signals. There are two main focuses about the work in this thesis: beamforming and DOA estimation. The original contributions include:

1. The first contribution is to improve the quaternion-valued capon beamformer. With crossed-dipole antenna arrays, quaternion-valued data models have been developed for beamforming for a long time. The quaternion-valued Capon beamformer is designed to eliminate the interference signals which are close to the signal of interest. The Q-capon beamformer tracks the aimed signals not only based on their directions, but also their polarisation information. However, for almost all quaternion-valued array models, especially for adaptive beamforming, the desired signal is still complex-valued. It leads to a problem that although the output signal of our designed beamformer is quaternion-valued, only two components of the quaternion result is fully made use of and the remaining two are simply discarded. As a result, significant redundancy exists in its implementation. To deal with the disadvantage of the original quaternion-valued beamformer, we consider a quaternion-valued desired signal and develop a fully quaternion-valued Capon beamformer, which has a better performance and a much lower complexity. Furthermore, based on this full quaternion model, the robust beamforming problem is also studied in the presence of steering vector errors and a worst-case based robust beamformer is developed. However, as shown in our later work, there are some interferences from some specific DOAs and polarisations cannot be suppressed effectively by the proposed method,

which is an inherent ambiguity issue associated with the array structure.

2. In the second contribution, the ambiguity problem for DOA estimation associated with vector sensor arrays is analysed. Electromagnetic (EM) vector sensor arrays are employed to track both the polarisation and direction of arrival (DOA) parameters of the impinging signals. However, the linear crossed-dipole arrays, due to inherent limitation of the structure, can only track one DOA parameter and two polarisation parameters. For full 4-D (2 DOA and 2 polarisation) estimation, we could extend the linear crossed-dipole array to the planar case. In our work, instead of extending the array geometry, we replace the crossed-dipoles by tripoles and construct a linear tripole array. A detailed derivation is given to show how a crossed-dipole array produces the ambiguity problem and why tripole arrays can avoid the problem. It is shown that the tripole sensor structure can estimate the 2-D DOA and 2-D polarisation information effectively. Besides, a MUSIC-based dimension-reduction algorithm is developed so that the 4-D estimation problem can be simplified to two separate 2-D estimation problems, significantly reducing the computational complexity of the solution. The Cramér-Rao Bound (CRB) is also deduced as a reference to evaluate the performance of the developed algorithm.

3. In the third and also the final contribution, we consider an emerging scenario for DOA estimation. Currently, most estimation work focuses on single signal transmission (SST) signals. However, to make full use of the degree of freedom in the spatial domain, dual signal transmission (DST) model signal can be employed in wireless communications. But there is rarely any work on DST signal estimation. Motivated to solve the problem, a new two-step estimator is proposed for DST signals, where an SST estimator and an DST estimator are constructed separately to estimate the SST signals and the DST signals. To extend the two-step solution to a more general case, a unified general estimator is derived to estimate the DOAs of all mixed signals. An analysis is provided to explain why this general estimator works for both kinds of signals.

1.3 Outline

The outline of the thesis is as follows:

In Chapter 2, basic concepts of polarisation, beamforming and DOA estimation are reviewed. A crossed-dipole sensor array is introduced to describe how polarised signals are received. For beamforming, a well-known Capon beamformer and its application with worst-case constraints are reviewed and some simulation results are provided to demonstrate its performance. For DOA estimation, the well-known MUSIC algorithm is reviewed and the algorithm is extended to 1-D, 2-D and 4-D estimation cases.

In Chapter 3, with polarised signals, a uniform crossed-dipole linear array is applied to beamforming and a fully quaternion-valued Capon beamformer is proposed. The proposed beamformer considers the desired signal as quaternion-valued and a detailed derivation of the weighting vector is presented. Moreover, the new proposed beamformer is extended to the worst-case based robust beamforming. Simulation results including beam patterns and output SINR are provided to verify the effectiveness of the proposed algorithms.

In Chapter 4, the ambiguity problem for DOA estimation associated with vector sensor arrays is analysed. By comparing the difference between linear crossed-dipole and tripole sensor array, it can be found that generally the linear crossed-dipole sensor array has the ambiguity problem while linear tripole sensor array can avoid the problem. Detailed proof is given as a explanation to the problem. After that, based on the tripole senor array, a complexity-reduction MUSIC algorithm is proposed, which reduce the orginal 4-D search to 2-D searches. The CRB (Cramér-Rao Bound) is derived to evaluate the performance of the new algorithm.

In Chapter 5, an emerging scenario for DOA estimation is considered, where the incoming signals include both the single-signal transmission (SST) and the dual-signal transmission (DST) signals. Traditional 4-D MUSIC algorithm fails to find the direction and polarisation of the DST signals. To solve the problem, by exploiting the additional information provided by DST signals, a two-step algorithm is firstly proposed. In the solution, a rank-reduction SST estimator is applied to find the DOA and polarisation of

the SST signals and then a specifically designed DST estimator is derived to estimate the DOA of DST signals. Furthermore, a general MST estimator is proposed to obtain the DOA parameters of the SST and the DST signals in one step. After that, the CRB (Cramér-Rao Bound) is derived to evaluate the performance of the proposed algorithms.

In Chapter 6, conclusions are drawn and some ideas for future work are introduced.

Chapter 2

Literature Review

2.1 Introduction to Polarization

Polarisation is used to describe the phenomenon that transverse waves are able to oscillate in different directions simultaneously. For transverse waves, the oscillating directions are always orthogonal to the wave propagating direction, which is the reason of polarisation in transverse waves transmission. While for longitudinal waves, the oscillating directions are identical with the wave propagating direction. As a result, there is no polarisation in longitudinal wave propagation. Typical longitudinal waves, for example, the sound waves propagating in liquid or gas, do not exhibit polarisation. Those transverse waves, such as the sound waves propagating in solid media, and transverse electromagnetic waves, exhibit polarisation with waves propagating. In this section, the discussion of polarisation is mainly focus on transverse electromagnetic waves.

Consider a transverse electromagnetic (TEM) wave propagating to an array with polarisation sensitive sensors, for example, crossed-dipole sensor array, which is shown in Fig. 2.1. The electric field $\vec{\boldsymbol{E}}$ and the magnetic field $\vec{\boldsymbol{B}}$ are both orthogonal to the wave propagating direction, which can be denoted as an incoming unit vector $\hat{\boldsymbol{r}}$. As the strength of electric field and magnetic field can not be measured directly, the sensors use the induced voltage gradient caused by these two fields to describe the strength instead. In the

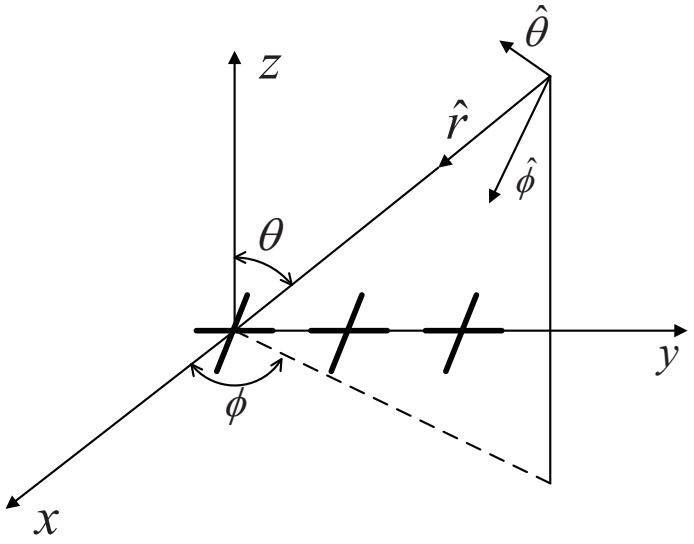

Figure 2.1: A linear crossed-dipole array.

following, the discussion will be based on a crossed-dipole sensor array. In crossed-dipole array, the information of magnetic field will be ignored and only the strength of electric field will be recorded in the array.

With the varying of time, the direction of electric field keeps changing. As the direction of electric field at any time-instant is always orthogonal to the wave propagating direction $\hat{\boldsymbol{r}}$, the directions of electric field lie in a planar. Here we use two polarized vectors $\hat{\boldsymbol{\phi}}$ and $\hat{\boldsymbol{\theta}}$ as two basic vectors to describe the electric field planar, where the two vectors must be both unit vectors and orthogonal to each other. Then, the electric field of the TEM wave can be expressed in terms of these vectors

$$\begin{aligned} \vec{\boldsymbol{E}} &= E_\phi \hat{\boldsymbol{\phi}} + E_\theta \hat{\boldsymbol{\theta}} + 0\hat{\boldsymbol{r}} \\ &= E_\phi \hat{\boldsymbol{\phi}} + E_\theta \hat{\boldsymbol{\theta}} \end{aligned} \tag{2.1}$$

Notice that the component in the $\hat{\boldsymbol{r}}$ direction is zero in (2.1), and the electric field of the wave is determined by the other two components. There are more than one choices in choosing $\hat{\boldsymbol{\phi}}$ and $\hat{\boldsymbol{\theta}}$. Generally we use the following equations in our analysis,

$$\hat{\boldsymbol{\phi}} = [-\sin\phi, \cos\phi, 0]^T \tag{2.2}$$

$$\hat{\boldsymbol{\theta}} = [\cos\phi\cos\theta, \cos\phi\sin\theta, -\sin\theta]^T \tag{2.3}$$

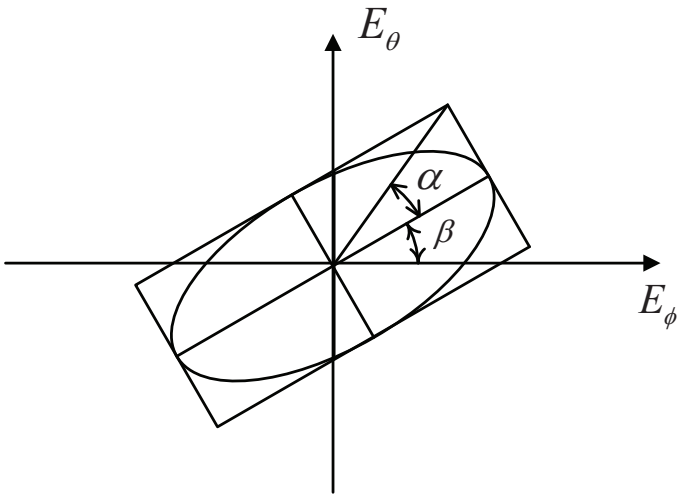

Figure 2.2: Polarization ellipse.

As time progresses, the amplitude of these two components varies periodically sharing a same frequency of the TEM wave. As a result, the electric field varies in a polarisation ellipse which is shown in Fig. 2.2. In the ellipse, we define β as the angle between the major axis of the ellipse and $\hat{\boldsymbol{\phi}}$. To eliminate ambiguities we also set β in the range $0 \leq \beta < \pi$. Another angle α is defined to describe the ellipticity, which is given by

$$\alpha = \tan^{-1} \frac{\text{minor} \quad \text{axis}}{\text{major} \quad \text{axis}} \tag{2.4}$$

α is defined positive when the electric ellipse rotates clockwise and negative when rotates counter-clockwise in the range of $-\pi/4 \leq \alpha \leq \pi/4$ [16]. For a given polarized signal, the electric field components are denoted by

$$E_\phi = A \cos\gamma \tag{2.5}$$

$$E_\theta = A \sin\gamma e^{j\eta} \tag{2.6}$$

A is the amplitude of the signal and the relationship among α, β, γ and η is

$$\cos 2\gamma = \cos 2\alpha \cos 2\beta \tag{2.7}$$

$$\tan\eta = \tan 2\alpha \csc 2\beta \tag{2.8}$$

where $0 \leq \gamma \leq \pi/4$ and $-\pi \leq \eta \leq \pi$.

Generally, the electric field is an ellipse from the view of wave propagating direction and we call it elliptical polarisation. However, the 'ellipse' may degenerate to a straight line or a circle. The two special cases are noted as linear polarisation and circular polarisation. When $\alpha = 0$, the electric field $\vec{\boldsymbol{E}}$ will be clustered in a line. While if $\alpha = \pm\frac{\pi}{4}$, in the ellipse, the length of minor axis will be equal to the length of major axis. As a result, the ellipse will appear as a circle.

When an elliptically polarised signal impinges on the crossed-dipole array (linearly polarised and circularly polarised signals can be considered as the special case of elliptically polarised signals), as we mentioned before, the strength of electric field is recorded in the form of induced voltage gradient and projected onto x, y and z axes in space, where

$$\begin{aligned}\vec{\boldsymbol{E}} &= E_\phi\hat{\boldsymbol{\phi}} + E_\theta\hat{\boldsymbol{\theta}} \\ &= (E_\theta\cos\theta\cos\phi - E_\phi\sin\phi)\hat{x} \\ &+(E_\theta\cos\theta\sin\phi + E_\phi\cos\phi)\hat{y} \\ &-(E_\theta\sin\theta)\hat{z} \end{aligned} \tag{2.9}$$

With more details to express E_ϕ and E_θ in (2.5), the electric field becomes

$$\begin{aligned}\vec{\boldsymbol{E}} &= A[(\sin\gamma\cos\theta\cos\phi e^{j\eta} - \cos\gamma\sin\phi)\hat{x} \\ &+(\sin\gamma\cos\theta\sin\phi e^{j\eta} + \cos\gamma\cos\phi)\hat{y} \\ &-(\sin\gamma\sin\theta e^{j\eta})\hat{z}] \\ &= A(p_x\hat{x} + p_y\hat{y} + p_z\hat{z}) \end{aligned} \tag{2.10}$$

where p_x, p_y, p_z are the polarisation coefficients in the Cartesian coordinate system.

In practical work, polarisation is an important design consideration. Although elliptical polarisation is the general case, the linear polarisation and circular polarisation are applied in communication systems more widely. For example, usually the linear polarisation are applied as vertical linear polarisation and vertical linear polarisation. Vertical linear polarisation is widely used in radiating the radio signals such as the mobile equipments. Horizontal linear polarisation is used to broadcast television signals, which can reduce the

interferences with mobile signals. Another consideration is that manmade noise usually appears as vertical linear polarisation, the horizontal linearly polarised signals suffer less interferences from noise.

Circular polarisation is mostly applied to satellite communications. One reason is that circular polarisation is more resistant to signal degradation due to the atmosphere conditions. For example, when signals travel through heavy clouds or rain, the rotation of the signals may change, which affects the linear polarisation more than the circular polarisation.

2.2 Introduction to Narrowband Beamforming

Beamforming is a technique in array signal processing to extract the signal of interest among interferences and background noise. In beamforming, the information of the signal envelope is collected in the array as a vector. Applying a weight vector to the received array signals, the final output includes the desired signal from a certain DOA, while the interferences with different DOAs and background noise are efficiently suppressed [17]. The main task of beamforming is to find a suitable weight vector. In the following, we will give a introduction about the low pass equivalent model for bandpass signals. Then, a narrowband model for the signal transmission will be presented. Finally, we will provide some concepts of Capon beamformer and robust Capon beamformer based on narrowband model.

2.2.1 Low pass equivalent model

In communication systems, the transmission signals are always real-valued. A typical real-valued band-pass signal $s(t)$ can be denoted as

$$s(t) = Re\{s_l(t)e^{j\omega_c t}\} \tag{2.11}$$

where $s_l(t)$ is a signal with zero center frequency and $e^{j\omega_c t}$ can be viewed as a complex-valued carrier signal. Thus, after Fourier transform, the frequency domain of $s(t)$ is given

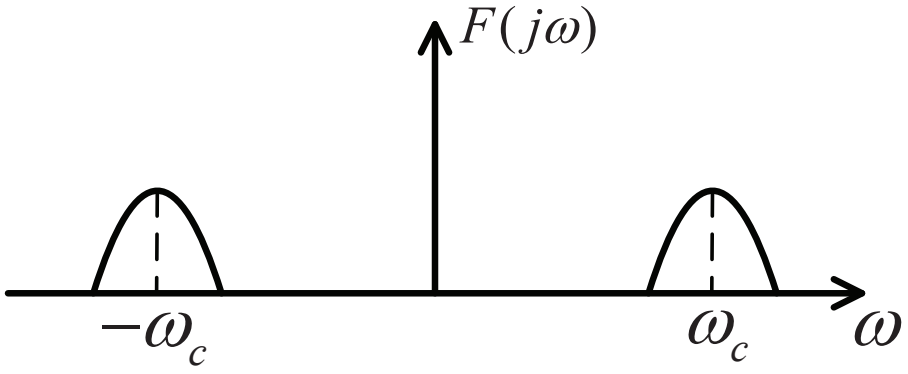

Figure 2.3: Frequency domain of a band-pass signal $s(t)$.

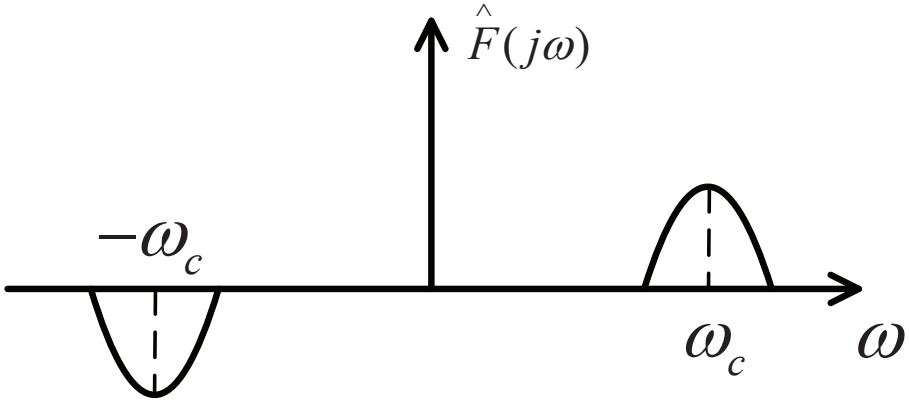

Figure 2.4: Frequency domain of signal $j\hat{s}(t)$.

in Fig. 2.3.

Due to the symmetry of the spectrum, the components with negative frequency can be discarded without any loss of the original signal information. Based on the idea, a new signal $j\hat{s}(t)$ can be added to the original signal to eliminate the components with negative frequency and double the components with positive frequency. The frequency domain of the new signal $j\hat{s}(t)$ is given in Fig. 2.4.

From Fig. 2.3 and Fig. 2.4, the relationship between $s(t)$ and $j\hat{s}(t)$ in frequency domain can be concluded as

$$\begin{aligned} j\hat{F}(\omega) &= \begin{cases} F(\omega), & \omega > 0 \\ 0, & \omega = 0 \\ -F(\omega), & \omega < 0 \end{cases} \\ &= sgn(\omega)F(\omega) \end{aligned} \tag{2.12}$$

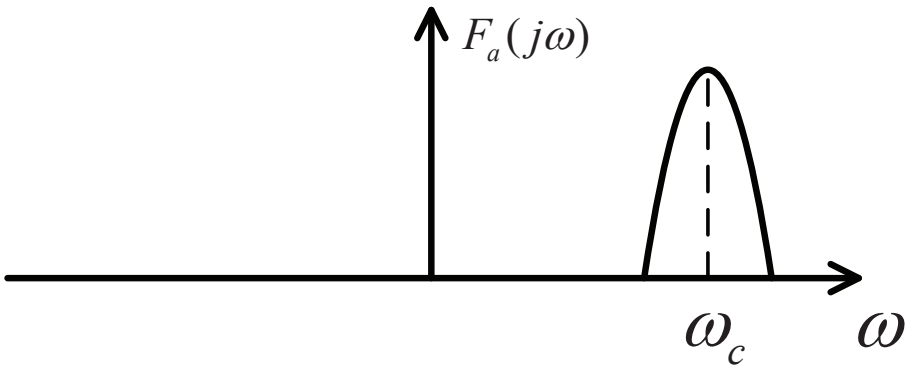

Figure 2.5: Frequency domain of analytic signal $s_a(t)$.

where the $sgn(\omega)$ is the sign function of ω. Apply the inverse Fourier transform, the relationship of the two signals in time domain is given by

$$\begin{aligned} j\hat{s}(t) &= \mathcal{F}^{-1}\{sgn(\omega)F(\omega)\} = \mathcal{F}^{-1}\{sgn(\omega)\} * \mathcal{F}^{-1}\{F(\omega)\} \\ &= j\frac{1}{\pi t} * s(t) \end{aligned} \tag{2.13}$$

where $*$ is the convolution operator. The signal $\hat{s}(t) = \frac{1}{\pi t} * s(t)$ is called as the Hilbert transform of the original signal $s(t)$. By adding the Hilbert transform to the original signal, the output signal $s_a(t)$ is a signal with only positive frequency. This signal is known as the analytic signal and

$$F_a(\omega) = F(\omega) + j\hat{F}(\omega) = 2F(\omega)u(\omega) \tag{2.14}$$

where $u(\omega)$ is the Heaviside step function. Since there is no components with negative frequency, the analytic signal is always complex-valued in time domain. In signal processing, as long as the functions applied to signals have no components of negative frequency, or the function is analytic, the original response can be obtained by simply discarding the imaginary part of analytic response. The frequency domain of the analytic signal is shown in Fig. 2.5.

In communication work, the output signal is the convolution of input signal and system response function. When the input signal $s(t)$ and the system impulse response $h(t)$ are bandpass functions, it requires high computation complexity to work out the output signal

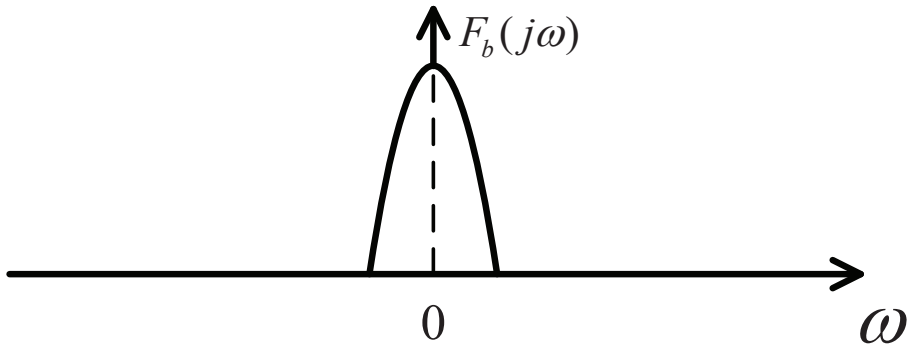

Figure 2.6: Frequency domain of low-pass equivalent signal $s_l(t)$.

$y(t)$. To reduce the work of computation, it's considerable to move the center frequency of the input signal and the system impulse response to 0. This is known as the low-pass equivalent model. The frequency domain of the low-pass equivalent signal $s_l(t)$ is given in Fig. 2.6. The Fourier transform of $s_l(t)$ can be denoted as

$$F_l(\omega) = F(\omega + \omega_c) + j\hat{F}(\omega + \omega_c) = 2F(\omega + \omega_c)u(\omega + \omega_c) \tag{2.15}$$

Similarly, the Fourier transform of the low-pass equivalent system impulse response $h_l(t)$ is

$$H_l(\omega) = H(\omega + \omega_c) + j\hat{H}(\omega + \omega_c) = 2H(\omega + \omega_c)u(\omega + \omega_c) \tag{2.16}$$

The output of low-pass equivalent model will be

$$Y_l(\omega) = F_l(\omega)H_l(\omega) = 4F(\omega + \omega_c)H(\omega + \omega_c)u(\omega + \omega_c) \tag{2.17}$$

If only keep the real-valued components of $y_l(t)$ with a $e^{j\omega_c t}$ phase shift in time domain, the Fourier transform is

$$\begin{aligned} Re\{y_l(t)e^{j\omega_c t}\} \leftrightarrow \quad & \frac{Y_l(\omega - \omega_c) + Y_l(-\omega - \omega_c)}{2} \\ & = \frac{4F(\omega)H(\omega)u(\omega) + 4F(\omega)H(\omega)u(-\omega)}{2} \\ & = 2F(\omega)H(\omega) \leftrightarrow 2y(t) \end{aligned} \tag{2.18}$$

The equation illustrates that to avoid the high computation of $y(t)$, it's more convenient to compute $y_l(t)$ with the low-pass equivalent model. This low-pass equivalent model is

widely used to evaluate the performance of bandpass signals and bandpass systems. To recover the real-valued output $y(t)$, we only need to add a phase shift $e^{j\omega_c t}$ to the low-pass equivalent output $y_l(t)$, discard the imaginary-valued components and halve the result.

Compared to real-valued signals, complex-valued signal (low-pass equivalent signals) simplifies the calculation and makes attributes of functions more accessible. Besides, it also makes some developments on the derivation of modulation and demodulation techniques, such as single-sideband modulation. This explains a fact that in real communication systems the transmission signals are real-valued, however, when analysing signals, the complex-valued analytic signal is a better alternative.

2.2.2 Narrowband model

Consider an array of L omnidirectional sensors receive M uncorrelated source signals of frequency f_0 from the far field. For signal s_i, each sensor records a result and the received results differ by the arriving time at the array sensors. The time delay taken by a plane wave of s_i from the direction θ_i in two adjacent sensors is given by

$$\tau(\theta_i) = \frac{dsin\theta_i}{c} \tag{2.19}$$

In a uniform linear array (ULA), every two adjacent sensors have the same distance d, the time delay for source signal s_i between adjacent sensors shares the same value τ. Assuming that s_i arrives at the first sensor at time 0. Then the delay for the signal arrives at the lth sensor can be given as

$$\tau_l(\theta_i) = (l-1)\tau(\theta_i) \tag{2.20}$$

The signal received at the first sensor due to the i-th source can be normally expressed in complex notation as

$$s_i(t) = m_i(t)e^{j2\pi f_0 t} \tag{2.21}$$

with $m_i(t)$ denoting the complex baseband signal.

For the i-th source signal, the signal received at the lth sensor can be expressed as

$$s_i(t-\tau_l(\theta_i)) = m_i(t-\tau_l(\theta_i))e^{j2\pi f_0(t-\tau_l(\theta_i))} \tag{2.22}$$

The narrowband assumption for array signal processing states that if the bandwidth of the signal is narrow enough and the array dimensions are small enough, the modulating function will roughly stay constant after the time delay in all sensors. That is to say, $m_i(t+\tau_l(\theta_i)) \approx m_i(t)$.

The received signal at the l-th sensor can be expressed as [13]

$$s_i(t-\tau_l(\theta_i)) \approx m_i(t)e^{j2\pi f_0(t-\tau_l(\theta_i))} = m_i(t)e^{j2\pi f_0 t}e^{-j2\pi f_0\tau_l(\theta_i)} = s_i(t)a_{il} \tag{2.23}$$

where $a_{il} = e^{-j2\pi f_0\tau_l(\theta_i)}$, representing the phase shift at the l-th sensor of the i-th signal. Also, these phase shifts can be considered as a vector $\boldsymbol{a_i}$ varying with θ_i, which is known as the steering vector and can be denoted as

$$\boldsymbol{a_i} = [a_{i1}, a_{i2}, ..., a_{in}] \tag{2.24}$$

Let $x_l(t)$ denote the total received signal due to all M sources and background noise at the l-th sensor. It is given by

$$x_l(t) = \sum_{i=1}^{M} s_i(t)a_{il} + n_l(t) \tag{2.25}$$

where $n_l(t)$ is a Gaussian white noise at the l-th sensor with zero mean and variance σ_n^2.

A narrowband beamformer is shown in Fig. 2.7. The received array signals are multiplied by a corresponding weight coefficient w_l. After that, all the weighted signals are summed to form a new array output. The output can be denoted as

$$y(t) = \sum_{i=1}^{L} w_l^* x_l(t) \tag{2.26}$$

If we denote the weights and signals received by all array sensors as vector $\boldsymbol{w}$ and $\boldsymbol{x}(t)$, then

$$\begin{aligned} \boldsymbol{w} &= [w_1, w_2, ..., w_L]^T \\ \boldsymbol{x}(t) &= [x_1(t), x_2(t), ..., x_L(t)]^T \end{aligned} \tag{2.27}$$

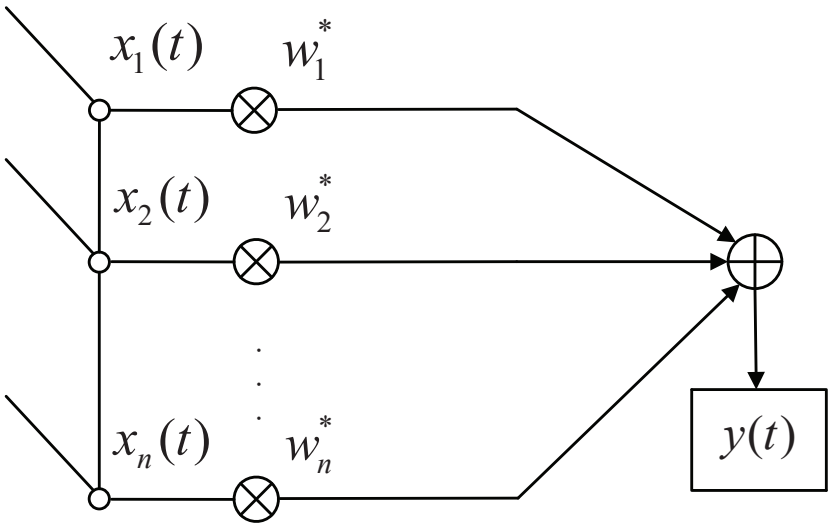

Figure 2.7: A general narrowband beamforming structure.

The output is given by

$$y(t) = \boldsymbol{w}^H \boldsymbol{x}(t) \tag{2.28}$$

where H denotes the Hermitian transpose of a matrix. $\boldsymbol{w}$ and $\boldsymbol{x}(t)$ are referred as array weight vector and array signal vector, respectively.

The mean output power of the beamformer is given by

$$P = E[y(t) \cdot y(t)^*] = \boldsymbol{w}^H \boldsymbol{R} \boldsymbol{w} \tag{2.29}$$

where E denotes the expectation operator and $\boldsymbol{R}$ is the covariance matrix defined as

$$\boldsymbol{R} = E[\boldsymbol{x}(t) \cdot \boldsymbol{x}(t)^H] \tag{2.30}$$

2.2.3 Capon beamformer

There are many schemes to determine the weights of a beamformer. One well-known algorithm is the Capon beamformer, or the minimum variance distortionless response beamformer (MVDR) [18, 19]. The main idea of Capon beamformer is to maximize the output SINR through maintaining a unity response to the desired signal and eliminating the interferences and noise as much as possible.

Given an L-element ULA, a desired signal and M-1 interferences impinge on the array. (We assume the M signals are uncorrelated with each other.) After beamforming, the output corresponding to the desired signal s_1 is $\boldsymbol{w}^H \boldsymbol{a}_1 s_1$, where

$$\boldsymbol{a}_1 = [1, e^{-j2\pi f_0 \tau_2(\theta_1)}, ..., e^{-j2\pi f_0 \tau_l(\theta_1)}] \tag{2.31}$$

To have a unity response to the desired signal, it should be satisfied that

$$\boldsymbol{w}^H \boldsymbol{a}_1 = 1 \tag{2.32}$$

Similarly, the output corresponding to the other M-1 interferences and noise is $\sum_{i=2}^{M} \boldsymbol{w}^H \boldsymbol{a}_i s_i + \boldsymbol{w}^H \boldsymbol{n}$. To suppress the interferences and noise effectively, we minimize the power of the final output. The power of interferences and noise at the output is given by the following expression:

$$\begin{aligned} P_{i+n} &= (\sum_{i=2}^{M} \boldsymbol{w}^H \boldsymbol{a}_i s_i + \boldsymbol{w}^H \boldsymbol{n}) * (\sum_{i=2}^{M} \boldsymbol{w}^H \boldsymbol{a}_i s_i + \boldsymbol{w}^H \boldsymbol{n})^H \\ &= \boldsymbol{w}^H (\sum_{i=2}^{M} \boldsymbol{a}_i E[s_i s_i^H] \boldsymbol{a}_i^H + \boldsymbol{n}\boldsymbol{n}^H) \boldsymbol{w} \\ &= \boldsymbol{w}^H \boldsymbol{R}_{i+n} \boldsymbol{w} \end{aligned} \tag{2.33}$$

$\boldsymbol{R}_{i+n}$ is the array covariance matrix for interferences plus noise, and

$$\boldsymbol{R}_{i+n} = \sum_{i=2}^{M} \boldsymbol{a}_i E[s_i s_i]^H \boldsymbol{a}_i^H + \sigma_n^2 \boldsymbol{I}_L \tag{2.34}$$

where $\boldsymbol{I}_L$ is the $L \times L$ identity matrix. The aim of the beamformer comes to

$$\begin{aligned} \min \quad & \boldsymbol{w}^H \boldsymbol{R}_{i+n} \boldsymbol{w} \\ \text{subject to} \quad & \boldsymbol{w}^H \boldsymbol{a}_1 = 1 \end{aligned} \tag{2.35}$$

In practice, the estimate of interferences and noise correlation matrix is usually not available. Instead, the covariance matrix $\boldsymbol{R}$ for the whole received array signals is used in determine the weights. Thus, the Capon beamformer turns to solve the following problem:

$$\begin{aligned} \min \quad & \boldsymbol{w}^H \boldsymbol{R} \boldsymbol{w} \\ \text{subject to} \quad & \boldsymbol{w}^H \boldsymbol{a}_1 = 1 \end{aligned} \tag{2.36}$$

Thus, the beamformer weights are determined to minimize the output power while maintaining unity response to the desired signal. The constraint ensures that the desired signal passes through the beamformer undistorted. In such a case, the desired signal's power remains the same at the output. Keeping the desired signal and minimizing the total output power ensure the maximum output SINR.

The Lagrange multipliers method can be applied to find the optimum value of weight vector $\boldsymbol{w}$, and the problem in (2.36) can be transformed into

$$\nabla_{\boldsymbol{w}^*}[\boldsymbol{w}^H\boldsymbol{R}\boldsymbol{w} - \lambda(\boldsymbol{w}^H\boldsymbol{a}_1 - 1)] = \boldsymbol{0} \tag{2.37}$$

where '∇' is the gradient symbol. The gradient of a function $l(\boldsymbol{w})$ with respect to a row vector $\boldsymbol{w}$ is defined as

$$\nabla_{\boldsymbol{w}} l(\boldsymbol{w}) = [\frac{\partial l(\boldsymbol{w})}{\partial w_1}, \frac{\partial l(\boldsymbol{w})}{\partial w_2}, ..., \frac{\partial l(\boldsymbol{w})}{\partial w_N}] \tag{2.38}$$

$w_1, w_2, ...w_N$ is the elements in vector $\boldsymbol{w}$. Notice that the result will have the same vector structure as $\boldsymbol{w}$. i.e, if $\boldsymbol{w}$ is a row or column vector, the gradient result will be a row or column vector with the same dimension as $\boldsymbol{w}$.

Here we divide the left side of (2.37) into two components $\nabla_{\boldsymbol{w}^*}\boldsymbol{w}^H\boldsymbol{R}\boldsymbol{w}$ and $\nabla_{\boldsymbol{w}^*}\lambda(\boldsymbol{w}^H\boldsymbol{a}_1 - 1)$. The first component can be computed as

$$\begin{aligned} \nabla_{\boldsymbol{w}^*}\boldsymbol{w}^H\boldsymbol{R}\boldsymbol{w} &= \nabla_{\boldsymbol{w}^*}[\boldsymbol{w}^H\boldsymbol{R}]\cdot\boldsymbol{w} + \boldsymbol{w}^H\boldsymbol{R}\nabla_{\boldsymbol{w}^*}\boldsymbol{w} \\ &= \boldsymbol{R}\boldsymbol{w} + \boldsymbol{w}^H\boldsymbol{R}\cdot 0 = \boldsymbol{R}\boldsymbol{w} \end{aligned} \tag{2.39}$$

The second component is

$$\nabla_{\boldsymbol{w}^*}\lambda(\boldsymbol{w}^H\boldsymbol{a}_1 - 1) = \lambda\boldsymbol{a}_1 \tag{2.40}$$

Then we have

$$\begin{aligned} & \boldsymbol{R}\boldsymbol{w} - \lambda\boldsymbol{a}_1 = \boldsymbol{0} \\ \Rightarrow \quad & \boldsymbol{w} = \lambda\boldsymbol{R}^{-1}\boldsymbol{a}_1 \end{aligned} \tag{2.41}$$

Substituting (2.41) into $\boldsymbol{w}^H\boldsymbol{a}_1 = 1$, λ is then give by

$$\lambda^* = \boldsymbol{a}_1^H\boldsymbol{R}^{-1}\boldsymbol{a}_1^{-1} \tag{2.42}$$

Note that λ is real-valued, then

$$\lambda = \lambda^* \tag{2.43}$$

We then obtain the final optimum weight vector $\boldsymbol{w}$ as

$$\boldsymbol{w} = (\boldsymbol{a}_1^H \boldsymbol{R}^{-1} \boldsymbol{a}_1)^{-1} \boldsymbol{R}^{-1} \boldsymbol{a}_1 \tag{2.44}$$

The beam response of the array for a signal from an arbitrary direction θ is defined as

$$p(\theta) = |\boldsymbol{w}^H \boldsymbol{a}(\theta)| \tag{2.45}$$

Next we give an analysis on the output SINR after Capon beamforming. The covariance matrix $\boldsymbol{R}$ includes two components, the part $\boldsymbol{R}_s$ for the signal of interest (SOI) and the part for interferences plus noise $\boldsymbol{R}_{i+n}$, which can be denoted as

$$\boldsymbol{R} = \boldsymbol{R}_s + \boldsymbol{R}_{i+n} \tag{2.46}$$

Assume there is only one desired signal s_1 and $M-1$ interference signals arriving at the array with N sensors ($M < N$). The output SINR is

$$SINR = \frac{\boldsymbol{w}^H \boldsymbol{R}_s \boldsymbol{w}}{\boldsymbol{w}^H \boldsymbol{R}_{i+n} \boldsymbol{w}} \tag{2.47}$$

$\boldsymbol{R}_s$ is calculated by the covariance of the steering vector of the desired signal, which is

$$\begin{aligned} \boldsymbol{R}_s &= \boldsymbol{a}_1 E[s_1 s_1^H] \boldsymbol{a}_1^H \\ &= \sigma_1^2 \boldsymbol{a}_1 \boldsymbol{a}_1^H \end{aligned} \tag{2.48}$$

Since the weight vector contains the part $\boldsymbol{R}^{-1}$, it is necessary to find out the inversion of the covariance matrix. The Matrix Inversion Lemma for an invertible matrix $\boldsymbol{A}$ and a vector $\boldsymbol{x}$ states that

$$(\boldsymbol{A} + \boldsymbol{x}\boldsymbol{x}^H)^{-1} = \boldsymbol{A}^{-1} - \frac{\boldsymbol{A}^{-1}\boldsymbol{x}\boldsymbol{x}^H\boldsymbol{A}^{-1}}{1 + \boldsymbol{x}^H\boldsymbol{A}^{-1}\boldsymbol{x}} \tag{2.49}$$

Since

$$\boldsymbol{R} = \sigma_s^2 \boldsymbol{a}_1 \boldsymbol{a}_1^H + \boldsymbol{R}_{i+n} \tag{2.50}$$

then it can be deduced that

$$\boldsymbol{R}^{-1} = \boldsymbol{R}_{i+n}^{-1} - \frac{\sigma_1^2 \boldsymbol{R}_{i+n}^{-1} \boldsymbol{a}_1 \boldsymbol{a}_1^H \boldsymbol{R}_{i+n}^{-1}}{1 + \sigma_1^2 \boldsymbol{a}_1^H \boldsymbol{R}_{i+n}^{-1} \boldsymbol{a}_1} \tag{2.51}$$

Substituting $\boldsymbol{R}^{-1}$ into (2.47), the output SINR is transformed into

$$SINR = \sigma_1^2 \boldsymbol{a}_1^H \boldsymbol{R}_{i+n}^{-1} \boldsymbol{a}_1 \tag{2.52}$$

A special scenario is that there is no other interference signals in the processing, and the output SNR becomes

$$\begin{aligned} SNR &= \sigma_1^2 \boldsymbol{a}_1^H \boldsymbol{R}_i^{-1} \boldsymbol{a}_1 \\ &= \frac{\sigma_1^2}{\sigma_n^2} \boldsymbol{a}_1^H \boldsymbol{a}_1 \\ &= N \frac{\sigma_1^2}{\sigma_n^2} \end{aligned} \tag{2.53}$$

After the beamforming process, the SNR increases by N times, which is the total number of elements of the array. With this situation, the inversion of covariance matrix is

$$\boldsymbol{R}^{-1} = \frac{\boldsymbol{I}}{\sigma_n^2} - \frac{\frac{\sigma_1^2}{\sigma_n^2} \boldsymbol{a}_1 \boldsymbol{a}_1^H}{\sigma_n^2 + N\sigma_1^2} \tag{2.54}$$

The weight vector becomes

$$\boldsymbol{w} = \frac{\boldsymbol{a}_1}{N} \tag{2.55}$$

Here we provide some simulation results to present how capon beamformer works and investigate the factors that affecting the beamforming performance. In the following simulation, the desired signal comes from 0° and the array sensor number is 6. The SNR is set to 0 dB. The distance between adjacent sensors equals half wavelength of our narrowband signals.

Firstly, we draw the beampattern of the Capon beamformer with no interfering signals. Fig. 2.8 gives the theoretical beam response for the range $\theta \in [-90^\circ, 90^\circ]$ with the snapshot number approaching infinity.

Then we consider the cases with finite number of snapshots. We give four cases with $100, 500, 1000$ and 5000 snapshots, respectively.

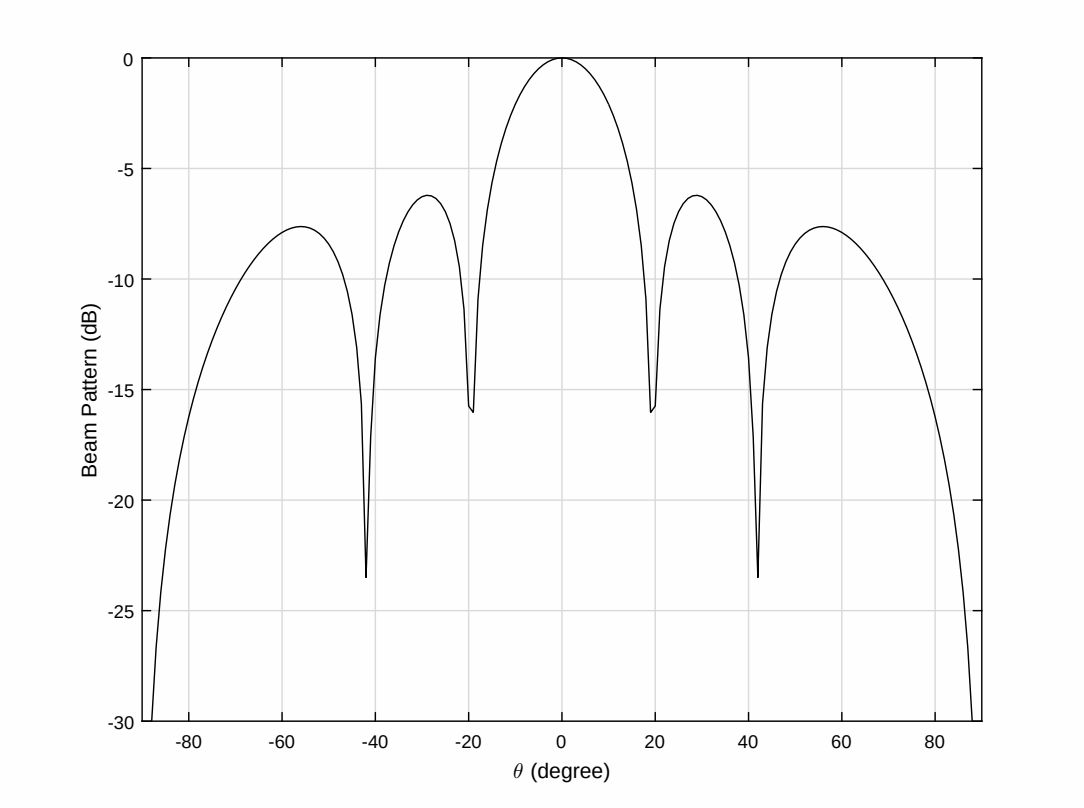

Figure 2.8: Beampattern with infinite number of snapshots, no interferences.

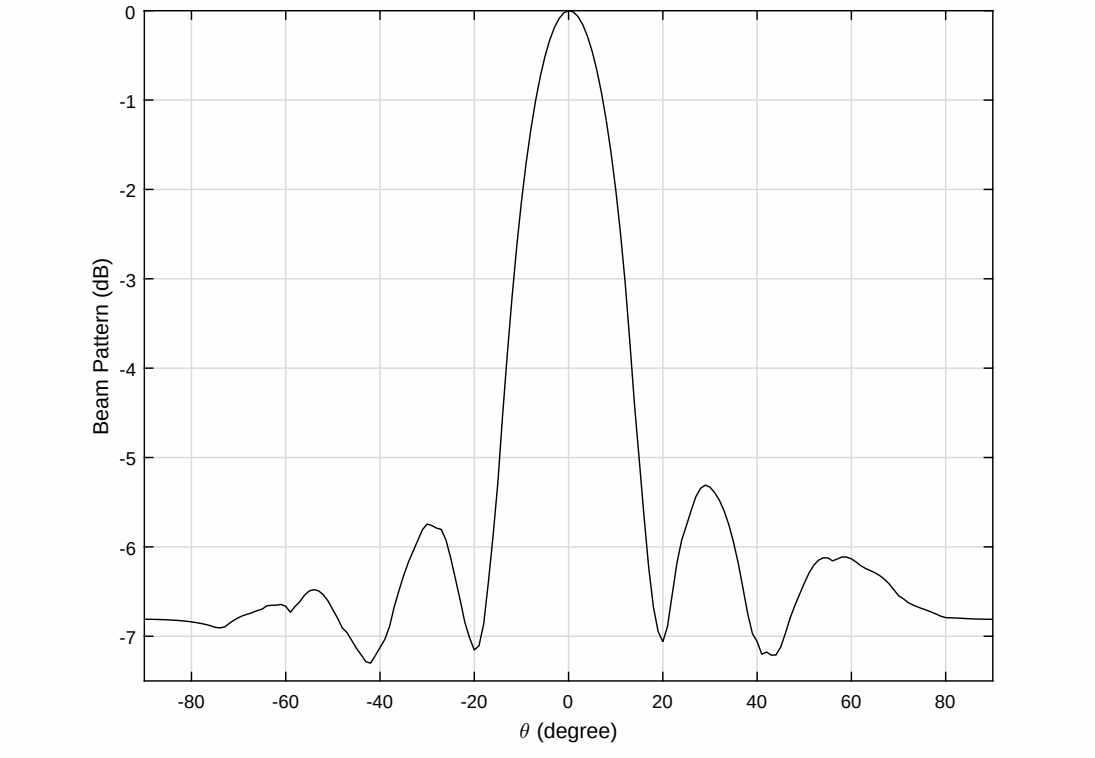

Figure 2.9: Beampattern with 100 snapshots, no interferences.

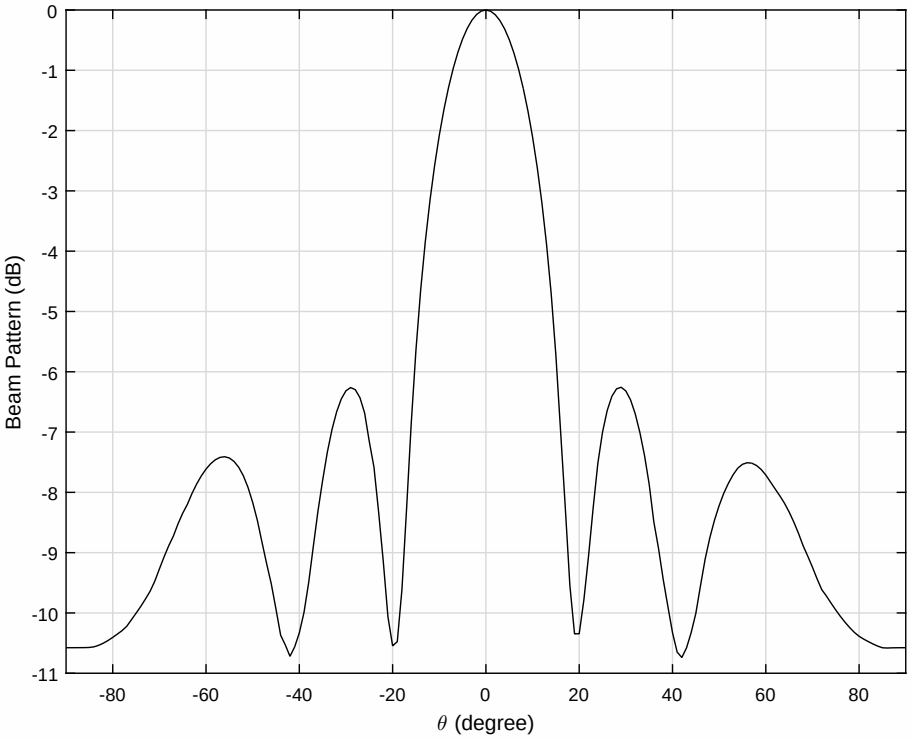

Figure 2.10: Beampattern with 500 snapshots, no interferences.

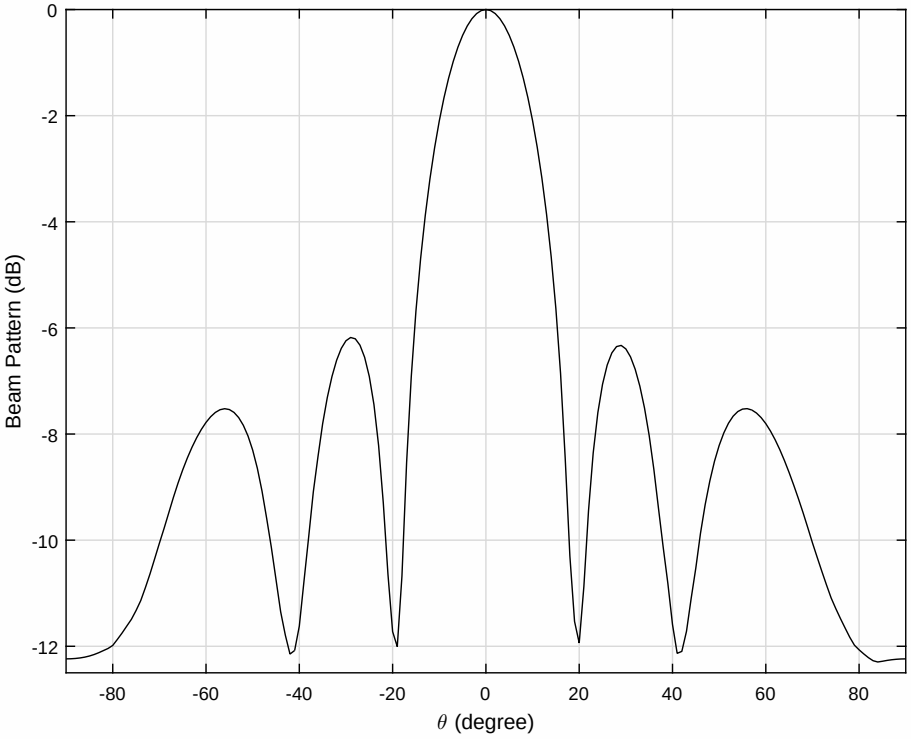

Figure 2.11: Beampattern with 1000 snapshots, no interferences.

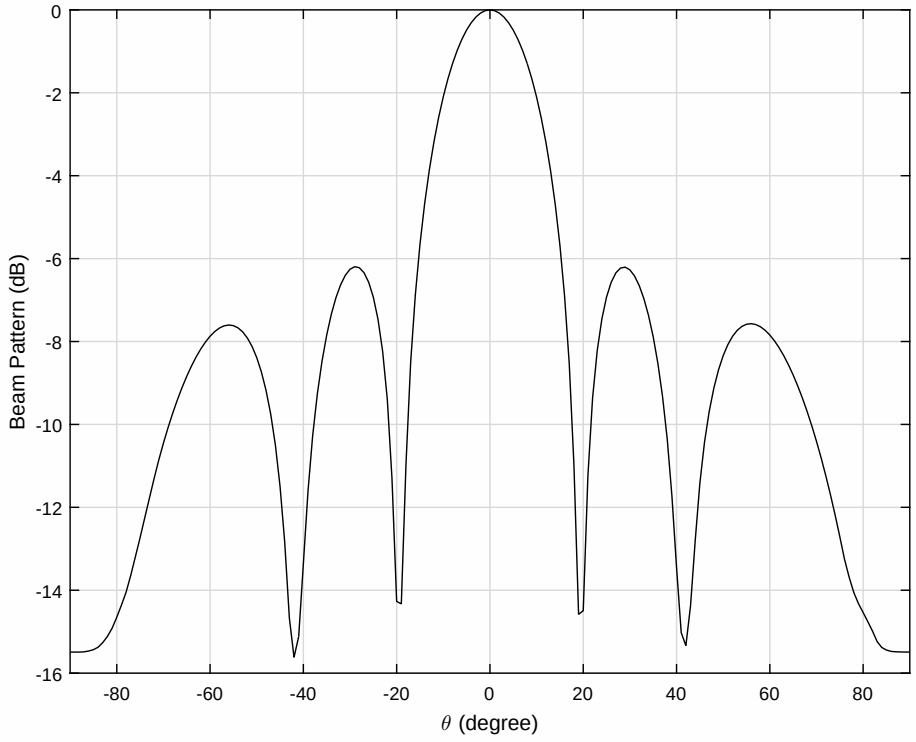

Figure 2.12: Beampattern with 5000 snapshots, no interferences.

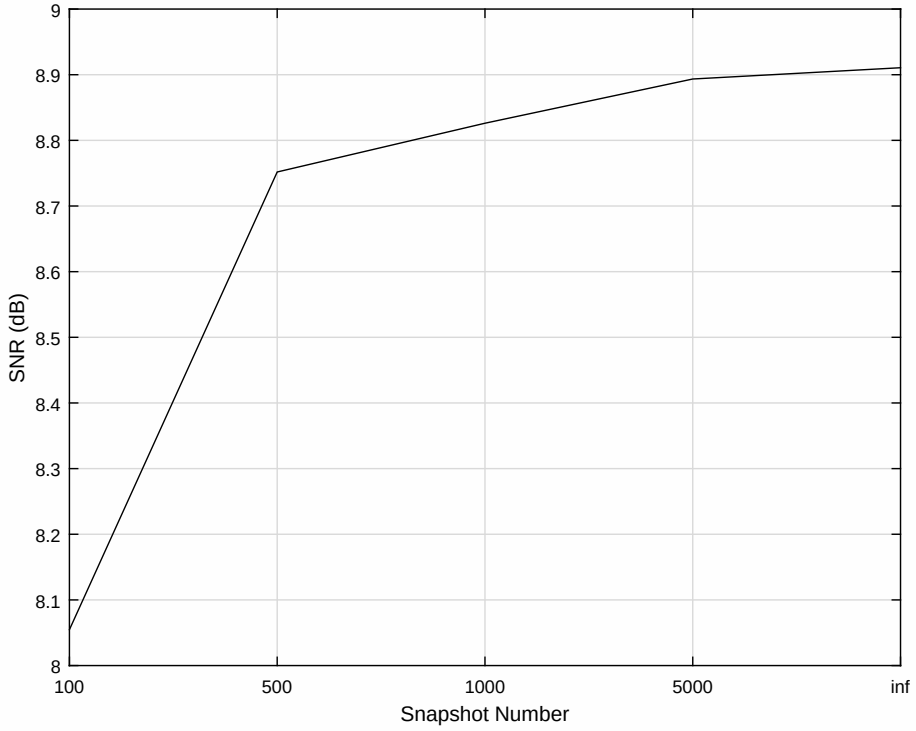

Figure 2.13: The output SNR versus snapshot numbers.

With the increase of snapshot numbers, the beampattern in Fig. 2.9-Fig. 2.12 becomes more and more close to the theoretical one. Fig. 2.13 also gives the output SNR comparison of the above cases.

2.2.4 Robust Capon beamformer

Although the Capon beamformer can recover the desired signal with improved SINR at its output. In practice, the assumed steering vector $\boldsymbol{a}_1$ may differ from the actual value for various reasons such as imprecise knowledge of the signal's direction and the sensor position errors. Many techniques for minimizing the beamforming errors have been proposed already, and one popular method is the robust capon beamformer, with worst-case constraints [20–25]. In the following we give a review of this technique.

In the robust Capon beamformer, we first replace the original constraint by the following one

$$\begin{aligned} \min \quad & \boldsymbol{w}^H \boldsymbol{R} \boldsymbol{w} \\ \text{subject to} \quad & \min |\boldsymbol{w}^H (\boldsymbol{a}_1 + \boldsymbol{e})| \geq 1 \end{aligned} \tag{2.56}$$

where $\boldsymbol{e}$ is the steering vector distortion caused by possible deviation from the actual direction, sensor position errors or other unknown factors. We consider the error has an upper norm bound ε so that $|\boldsymbol{e}| \leq \varepsilon$. The constraint can be further changed to

$$|\boldsymbol{w}^H (\boldsymbol{a}_1 + \boldsymbol{e})| = |\boldsymbol{w}^H \boldsymbol{a}_1 + \boldsymbol{w}^H \boldsymbol{e}| \geq |\boldsymbol{w}^H \boldsymbol{a}_1| - |\boldsymbol{w}^H \boldsymbol{e}| \tag{2.57}$$

Since $|\boldsymbol{e}| \leq \varepsilon$, then

$$|\boldsymbol{w}^H (\boldsymbol{a}_1 + \boldsymbol{e})| \geq |\boldsymbol{w}^H \boldsymbol{a}_1| - \varepsilon |\boldsymbol{w}| \tag{2.58}$$

With the constraints in (2.56), it can be deduced that

$$\begin{aligned} \min \quad & |\boldsymbol{w}^H (\boldsymbol{a}_1 + \boldsymbol{e})| = |\boldsymbol{w}^H \boldsymbol{a}_1| - \varepsilon |\boldsymbol{w}| \geq 1 \\ & |\boldsymbol{w}^H \boldsymbol{a}_1| \geq 1 + \varepsilon |\boldsymbol{w}| \end{aligned} \tag{2.59}$$

With the rotation of $\boldsymbol{w}$, we can always find another $\boldsymbol{w}_1 = \boldsymbol{w}e^{-j\phi}$ which keeps $\boldsymbol{w}_1^H \boldsymbol{R}\boldsymbol{w}_1$ unchanged and ensures $\boldsymbol{w}_1^H(\boldsymbol{a}_1 + \boldsymbol{e})$ real-valued.

Hence, the whole formulation changes to

$$\begin{aligned} \min \quad & \boldsymbol{w}^H \boldsymbol{R}\boldsymbol{w} \\ \text{subject to} \quad & |\boldsymbol{w}^H \boldsymbol{a}_1| \geq 1 + \varepsilon|\boldsymbol{w}| \end{aligned} \tag{2.60}$$

To find a solution for (2.60), we apply the CVX tool in MATLAB. Here, we give some simulation results for the above robust capon beamformer.

Firstly, consider a scenario where the desired signal comes from $\theta = 1^\circ$ with $SNR = 0$ dB. However, we have an imprecise knowledge of the direction as we assume that the direction of the signal is from $\theta = 0^\circ$. In the simulation, the results are obtained by 200 Monte-Carlo trials and we take 100 snapshots for each trial, and the error upper bound ε is set to 0.1. Fig. 2.14 is the beampattern of the robust Capon beamformer, where the response for the rough DOA range $[-3^\circ, 3^\circ]$ is beyond 0 dB. Moreover, Fig. 2.15 gives the output SINR comparison between the traditional Capon beamformer and robust Capon beamformer versus the input SINR.

2.3 Introduction of DOA Estimation

To estimate the direction of signal emitter by a sensor array is of great interest in many application areas such as radar, sonar, tomography and earthquake prediction. Many methods has been developed to solve the estimation problem. A classical method is to apply FFT (fast Fourier transform) to estimate DOA. However, when estimating multiple targets, the algorithm performs with low resolution that the estimation results may be not as accurate as required. To overcome the resolution problem, subspace method such as the multiple signal classification (MUSIC) [26] and estimating signal parameters via rotational invariance techniques (ESPRIT) [27] is developed. One well-known DOA estimation method, the MUSIC algorithm, will be introduced in detail in this section and the algorithm is the main tool exploited for DOA estimation in our research work.

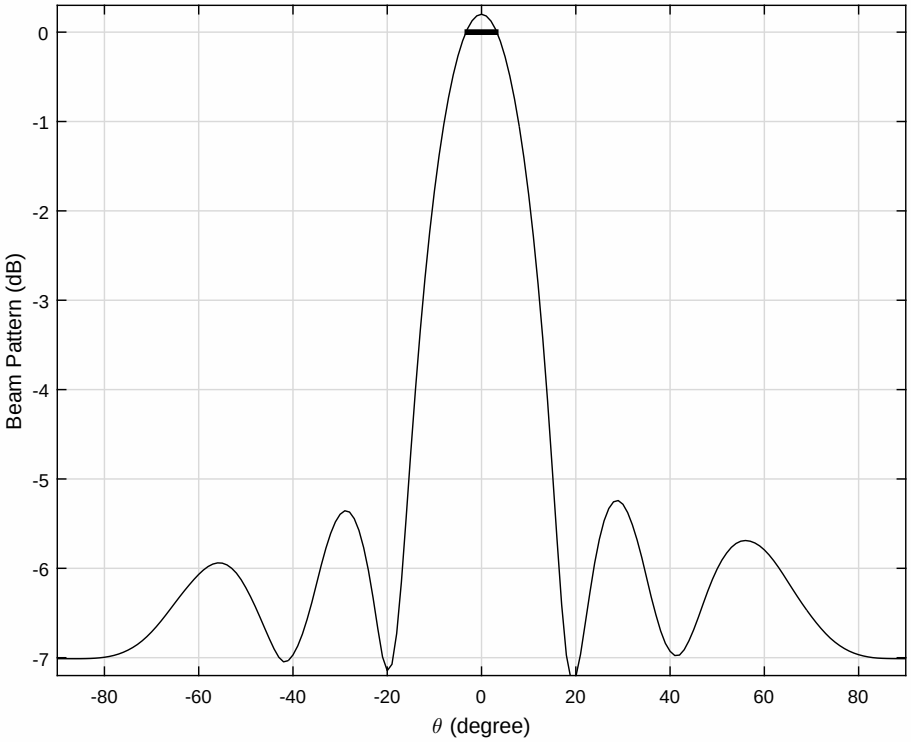

Figure 2.14: Beampattern of the robust Capon beamformer.

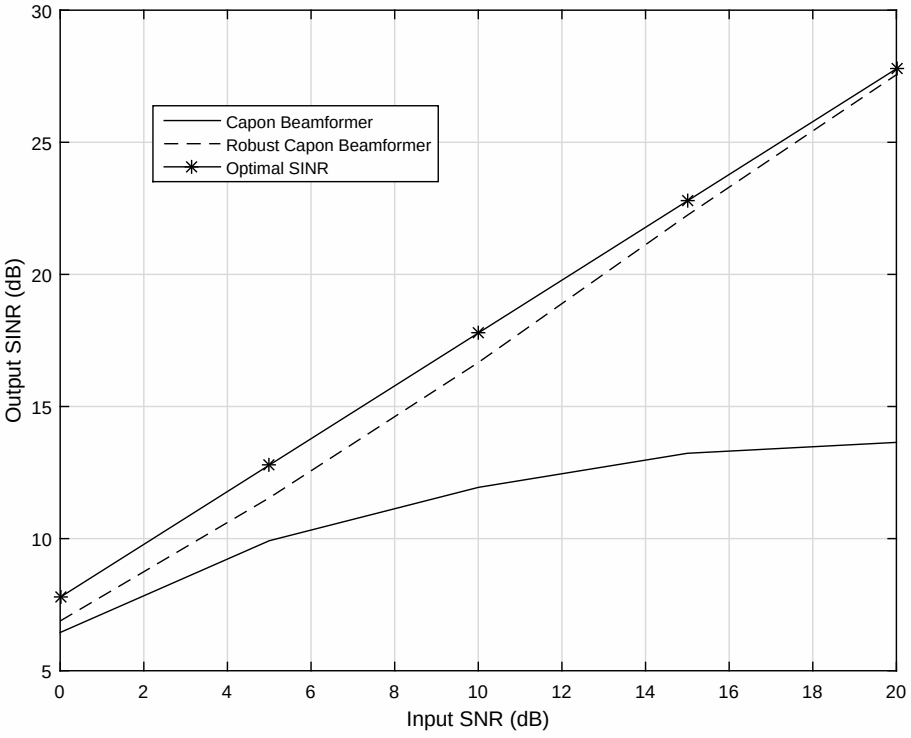

Figure 2.15: Output SINR performance comparison.

The MUSIC algorithm is an efficient method to estimate DOAs based on eigenstructure analysis. It provides the parameters of multiple signals arriving at a sensor array from the measurements at array elements. By analysing the received signals, parameters of the signals, including the direction, frequency and polarisation can be estimated. Based on different manifolds of the array and the model of the signals, the algorithm achieves different estimation dimensions. In the following, we will give a brief introduction about the classical FFT method. Then, MUSIC algorithm will be discussed as an example of subspace method. Simulation results are given to show the different resolution of the two methods. Finally, we extend the MUSIC algorithm in specific scenarios.

2.3.1 FFT method

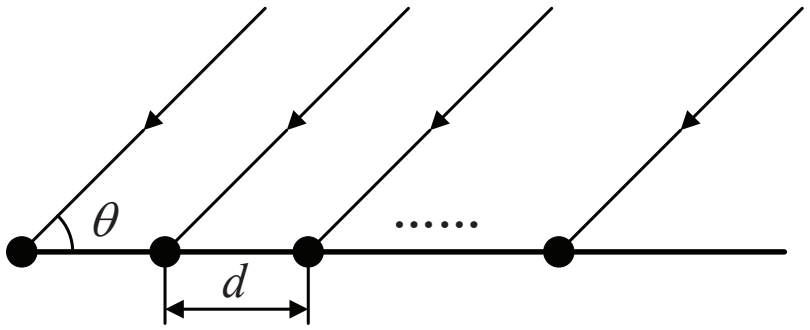

Figure 2.16: The structure of a uniform linear array.

Consider a ULA (Fig. 2.16) with M elements receiving N signals from different directions. Similar to the model in beamforming, the sensor array collects received signals at each sensor to form a data vector $\boldsymbol{x}(t)$. The data model can be described as [26, 28]

$$\boldsymbol{x}(t) = [\boldsymbol{a}_1, \boldsymbol{a}_2, ..., \boldsymbol{a}_N][s_1(t), s_2(t), ..., s_N(t)]^T + \boldsymbol{n} = \boldsymbol{A}\boldsymbol{s}(t) + \boldsymbol{n} \tag{2.61}$$

with

$$\boldsymbol{a}_i = [1, e^{\frac{-j2\pi d\theta_i}{\lambda}}, ..., e^{\frac{-j(M-1)2\pi d\theta_i}{\lambda}}]^T \tag{2.62}$$

where λ is the wavelength of the incoming signals and $\boldsymbol{a}_i$ ($i \in [1, N]$) is the steering vector associated with the 1-D direction parameter θ_i. The received array signal vector $\boldsymbol{x} \in C^{M\times 1}$. The incoming signals are represented with their amplitude and phase information by complex values s_i ($i \in [1, N]$). Noise in each sensor element, whether generated by the internal instrumentation circuits or induced along with signals, appears in the vector $\boldsymbol{n} \in C^{M\times 1}$. If we ignore the noise, after a weighted vector $\boldsymbol{w}^H \in C^{q\times M}$ in Fig. 2.7, the received signal $y(t)$ can be denoted as

$$y(t) = \boldsymbol{w}^H \boldsymbol{x}(t) = \boldsymbol{w}^H \boldsymbol{A}\boldsymbol{s}(t) \tag{2.63}$$

where $\boldsymbol{w}^H$ is a searching vector that

$$\boldsymbol{w}^H(\theta) = [1, e^{\frac{j2\pi d\theta}{\lambda}}, ..., e^{\frac{j2\pi(M-1)d\theta}{\lambda}}] \tag{2.64}$$

When the searching angle θ meets the impinging DOA θ_i, $\boldsymbol{w}^H \boldsymbol{a}_i$ will be maximized and then $y(t)$ will reach a local maximum. Hence, we can define a spectrum function to obtain target directions by

$$F(\theta) = |y(t, \theta)| = |\boldsymbol{w}^H(\theta)\boldsymbol{x}(t)| \tag{2.65}$$

Notice that

$$y(t, \theta) = \boldsymbol{w}^H(\theta)\boldsymbol{x}(t) = \sum_{i=1}^{M} x_i(t) e^{\frac{j2\pi(i-1)d\theta}{\lambda}} \tag{2.66}$$

This means $y(t, \theta)$ is the fast Fourier transform (FFT) of $\boldsymbol{x}(t)$ and the spectrum function $F(\theta)$ can be also obtained by applying FFT. We give some estimation results by FFT method in Fig. 2.17 - 2.20. It can be obtained that with the increase of sensor numbers and SNR, the estimation performances also increase. However, FFT method shows low resolution in distinguishing different angles. Besides, since the math model considers the received signal without background noise, the results are very sensitive with SNR. With low SNR, the FFT method may get very bad estimating results.

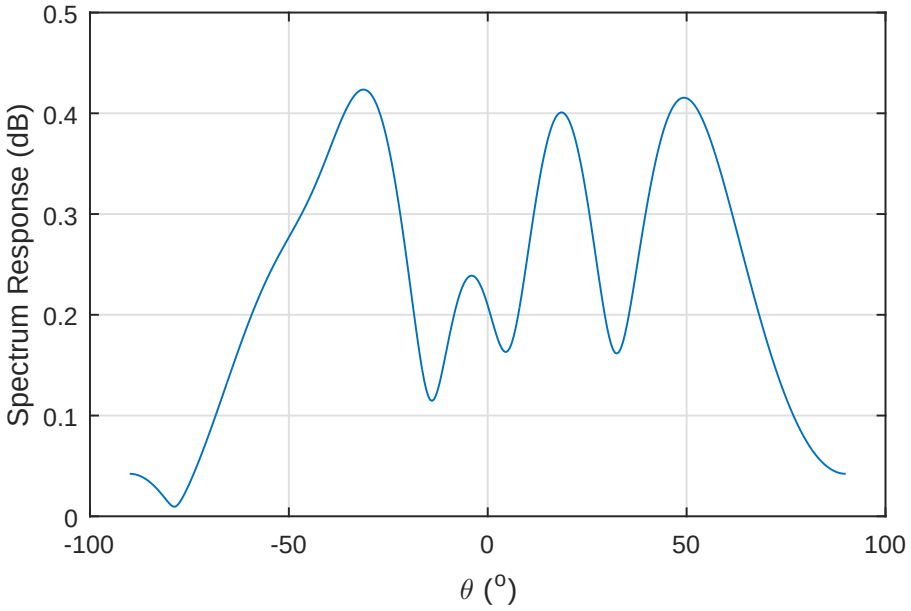

Figure 2.17: FFT method with 2 sources from $-40°$ and $60°$, 100 snapshots, 6 sensors, and SNR = 0 dB.

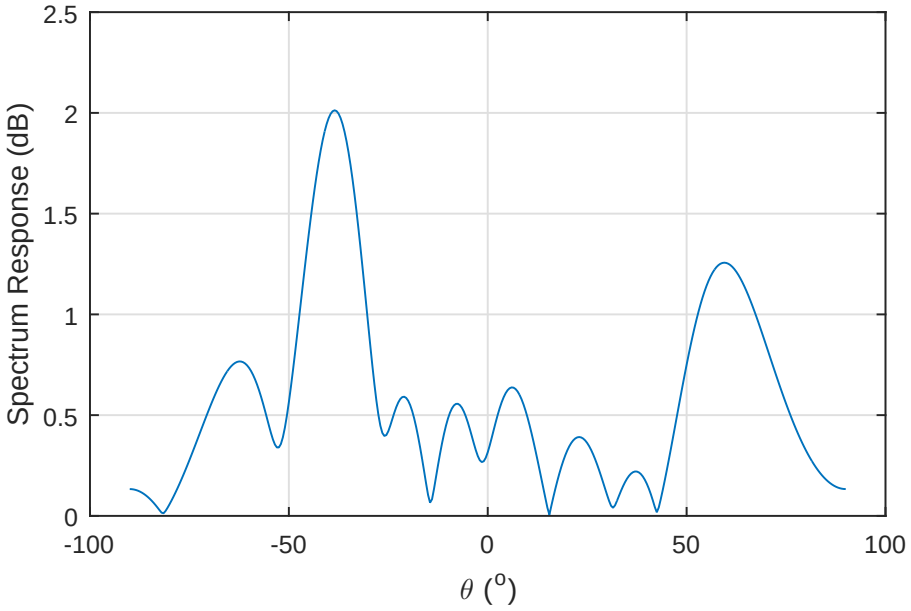

Figure 2.18: FFT method with 2 sources from $-40°$ and $60°$, 100 snapshots, 10 sensors, and SNR = 0 dB.

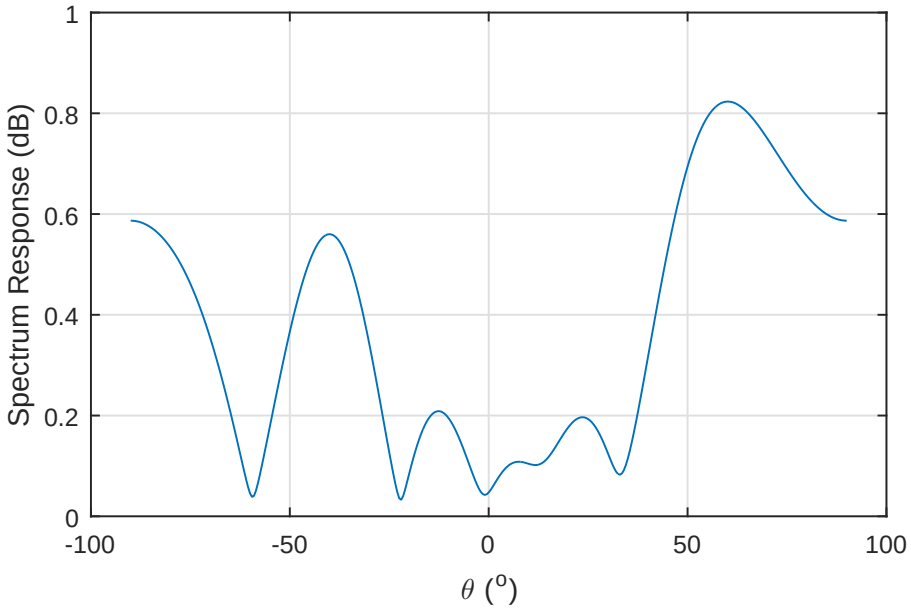

Figure 2.19: FFT method with 2 sources from $-40°$ and $60°$, 100 snapshots, 6 sensors, and SNR = 10 dB.

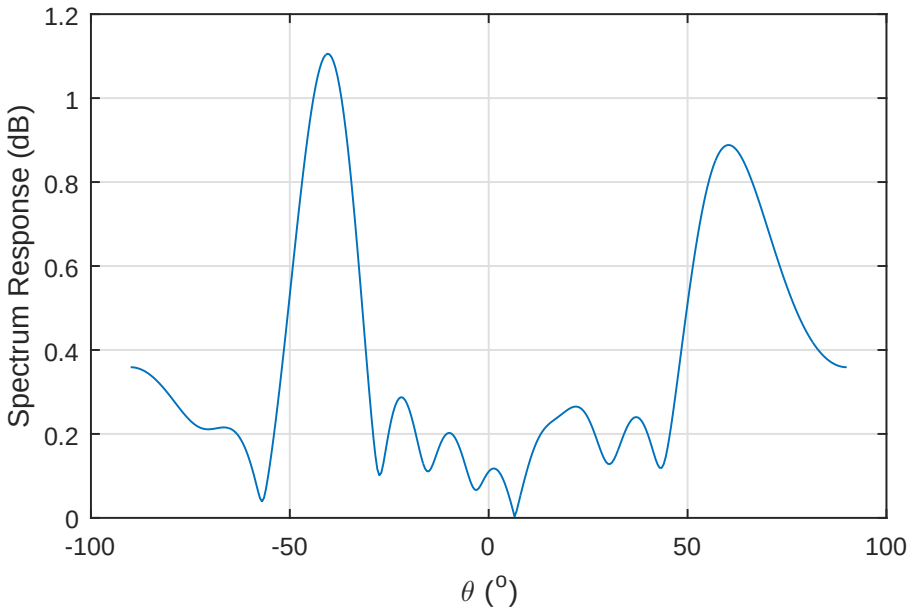

Figure 2.20: FFT method with 2 sources from $-40°$ and $60°$, 100 snapshots, 10 sensors, and SNR = 10 dB.

2.3.2 1-D MUSIC with ULA

Similar with the signal model in FFT method, the signal received at the array can be denoted as

$$\boldsymbol{x}(t) = \boldsymbol{A}\boldsymbol{s}(t) + \boldsymbol{n} \tag{2.67}$$

Instead of the weighted vector, we construct the covariance matrix $\boldsymbol{R}_x$ of $\boldsymbol{x}$, which can be denoted as

$$\begin{aligned} \boldsymbol{R}_x &= E[\boldsymbol{x}\boldsymbol{x}^H] = \boldsymbol{A}E[\boldsymbol{s}(t)\boldsymbol{s}^H(t)]\boldsymbol{A}^H + E[\boldsymbol{n}\boldsymbol{n}^H] \\ &= \boldsymbol{R}_s + \boldsymbol{R}_n \end{aligned} \tag{2.68}$$

where $\boldsymbol{R}_s$ is the covariance matrix of signals, expressed by

$$\boldsymbol{R}_s = \boldsymbol{A}E[\boldsymbol{s}(t)\boldsymbol{s}^H(t)]\boldsymbol{A}^H \tag{2.69}$$

$\boldsymbol{R}_n$ is the covariance matrix of noise, expressed by

$$\boldsymbol{R}_n = E[\boldsymbol{n}\boldsymbol{n}^H] = \sigma_n^2 \boldsymbol{I}_M \tag{2.70}$$

Applying eigenvalue decomposition (EVD), the signal covariance matrix $\boldsymbol{R}_s$ can be decomposed into

$$\boldsymbol{R}_s = \sum_{k=1}^{M} \alpha_k \boldsymbol{u}_k \boldsymbol{u}_k^H \tag{2.71}$$

where $\boldsymbol{u}_k$ is the k-th eigenvector and α_k is the corresponding eigenvalues (in descending order). When the number of source signals N is less than M, $\boldsymbol{R}_s \in \mathbb{H}^{M\times M}$ will not be a full rank matrix with $rank(\boldsymbol{R}_s) = N$, and it only has N non-zero eigenvalues, which means $\alpha_1 \geq \alpha_2 \geq \cdots \alpha_N \geq \alpha_{N+1} = \cdots = \alpha_M = 0$.

Since $\boldsymbol{R}_n = \sigma^2 \boldsymbol{I}_M$, we can write $\boldsymbol{R}_n$ in terms of

$$\boldsymbol{R}_n = \sum_{k=1}^{M} \sigma_n^2 \boldsymbol{u}_k \boldsymbol{u}_k^H \tag{2.72}$$

Then

$$\boldsymbol{R}_x = \boldsymbol{R}_s + \boldsymbol{R}_n = \sum_{k=1}^{M} \lambda_k \boldsymbol{u}_k \boldsymbol{u}_k^H \tag{2.73}$$

where λ_k is the k-th eigenvalue of the covariance matrix $\boldsymbol{R}_x$ satisfying $\lambda_k = \alpha_k + \sigma_n^2$. As $\boldsymbol{R}_s$ is the function of steering matrix $\boldsymbol{A}$, any steering vector $\boldsymbol{a}_i$ in $\boldsymbol{A}$ should be the linear combination of the eigenvectors with non-zero eigenvalues in $\boldsymbol{R}_s$, which are the also the first N eigenvectors in $\boldsymbol{R}_x$. These eigenvectors construct an N-dimension matrix, which is referred as the signal subspace.

The remaining $M - N$ eigenvectors of $\boldsymbol{R}_x$ have the same eigenvalue σ_n^2. Since $\boldsymbol{R}_x$ is Hermitian, the eigenvectors with different eigenvalues are orthogonal to each other, which means the the arbitrary steering vector $\boldsymbol{a}_i$ is orthogonal to the last $M - N$ eigenvectors in $\boldsymbol{R}_x$. Denoting the matrix including the last $M - N$ eigenvectors by $\boldsymbol{U}_N$, which is referred as the noise subspace, then

$$\boldsymbol{U}_N^H \boldsymbol{a}_i = \boldsymbol{0} \tag{2.74}$$

The minimum repeated eigenvalues can be found by computing all the eigenvalues of $\boldsymbol{R}_x$, the noise subspace is constructed by eigenvectors related to these eigenvalues. Once the noise subspace is estimated, the work next is to search for the DOA of the M incoming signals. Foe convenience, we ignore the label i of $\boldsymbol{a}_i$ and let $\boldsymbol{a}(\theta)$ denote the steering vector corresponding to direction θ and then search through θ from $-90°$ to $90°$ with a step size, for example $0.5°$. This can be accomplished by searching for the peaks of the function

$$P_{MU}(\theta) = \frac{1}{\boldsymbol{a}(\theta)^H \boldsymbol{U}_N \boldsymbol{U}_N^H \boldsymbol{a}(\theta)} \tag{2.75}$$

The M peaks in the function above provide the information of DOA of M incoming signals. Now we give a DOA spatial spectrum with the 1-D MUSIC in Fig. 2.21.

2.3.3 2-D MUSIC with URA

The 1-D MUSIC estimates the angle between the signal direction and the sensor array. Actually the method only detects the possible cone range of the source signals, which is

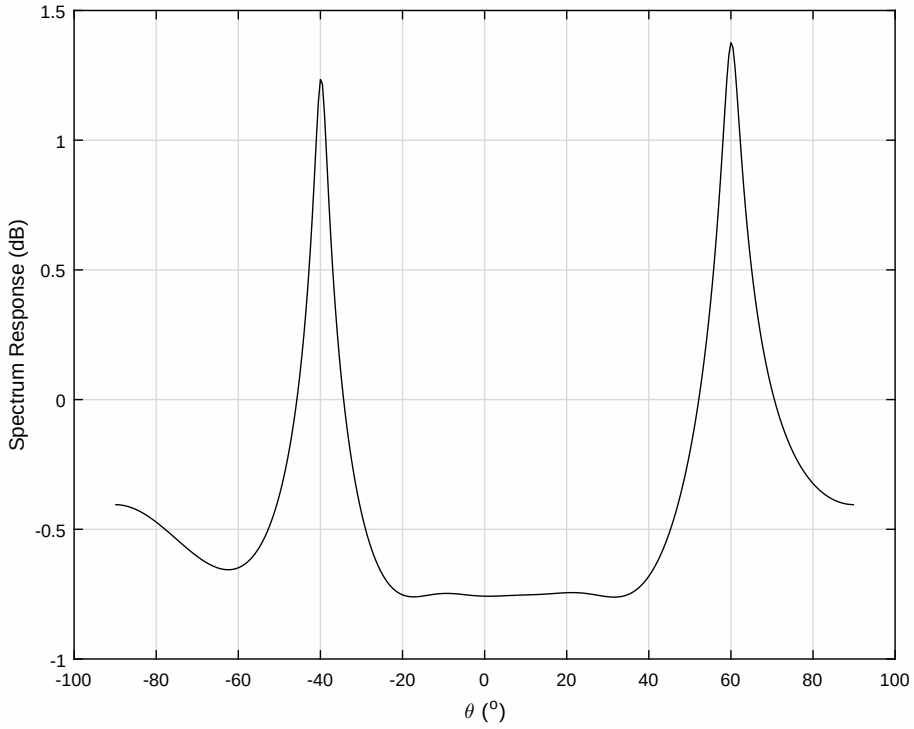

Figure 2.21: 1-D MUSIC with 2 sources from $-40°$ and $60°$, 100 snapshots, and SNR $= 0$ dB.

shown in Fig. 2.22. However, in some situations, the azimuth-elevation angle information need to be estimated. To achieve a 2-D DOA estimation, one solution is to extend the linear array into a planar array. One of the basic planar array structures is the uniform rectangular array (URA) [29–32], which is shown in Fig. 2.23. In this model, the steering vector $\boldsymbol{a}_n$ is associated with two direction parameters: the azimuth angle θ_n and the elevation angle ϕ_n, which is denoted by

$$\boldsymbol{a}(\theta_i, \phi_i) = \boldsymbol{a}_x(\theta_i, \phi_i) \otimes \boldsymbol{a}_y(\theta_i, \phi_i) \tag{2.76}$$

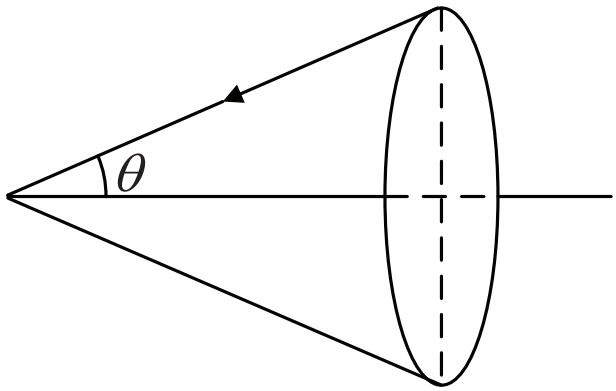

Figure 2.22: Cone range of source signals for linear arrays.

The symbol '$\otimes$' is the Kronecker product. $\boldsymbol{a}_x(\theta_i, \phi_i)$ and $\boldsymbol{a}_y(\theta_i, \phi_i)$ describe the phase shift along the x-axis direction and y-axis direction, respectively. The expression is given by

$$\begin{aligned}
\boldsymbol{a}_x(\theta_i, \phi_i) &= [1, e^{-j\frac{2\pi dx}{\lambda}\sin\theta_i \sin\phi_i}, ..., e^{-j\frac{2\pi(N-1)dx}{\lambda}\sin\theta_i \sin\phi_i}] \\
\boldsymbol{a}_y(\theta_i, \phi_i) &= [1, e^{-j\frac{2\pi dy}{\lambda}\sin\theta_i \cos\phi_i}, ..., e^{-j\frac{2\pi(N-1)dy}{\lambda}\sin\theta_i \cos\phi_i}]
\end{aligned} \tag{2.77}$$

Similar to the 1-D MUSIC algorithm, the 2-D DOA estimation can be achieved by searching for the peaks in the following cost function

$$P_{MU}(\theta, \phi) = \frac{1}{\boldsymbol{a}(\theta, \phi)^H \boldsymbol{U}_N \boldsymbol{U}_N^H \boldsymbol{a}(\theta, \phi)} \tag{2.78}$$

An example for 2-D MUSIC search result is provided in Fig. 2.24 and Fig. 2.25.

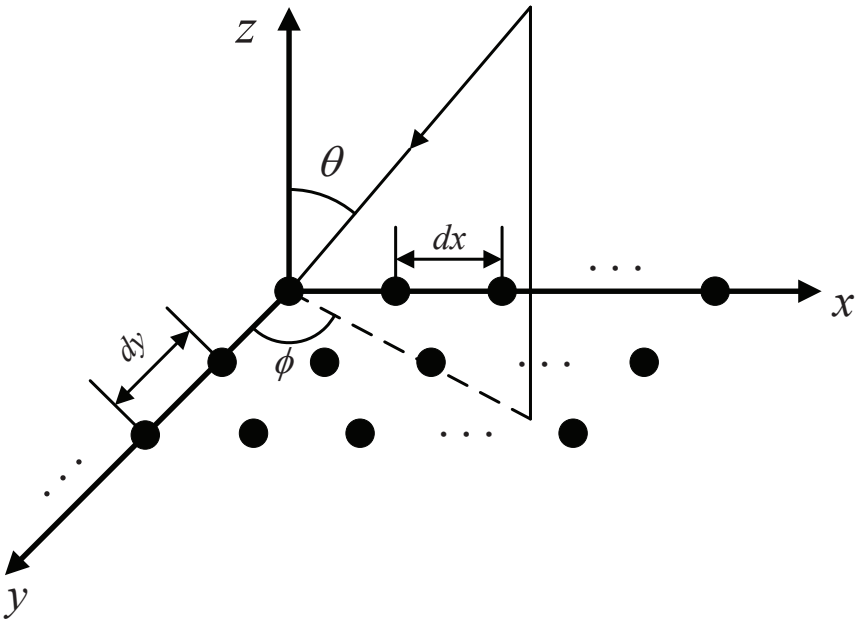

Figure 2.23: Structure of a uniform rectangular array.

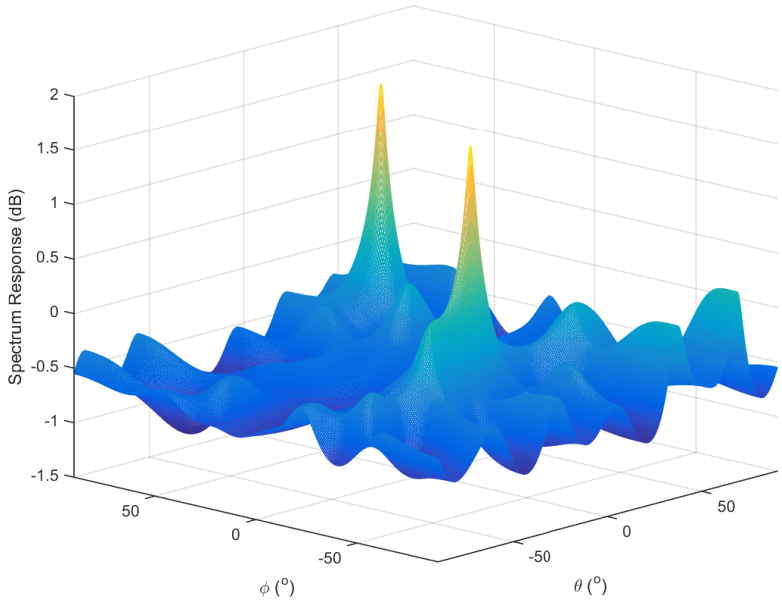

Figure 2.24: DOA spectrum of 2-D MUSIC, with two sources from $(\theta_1, \phi_1) = (60°, 40°)$ and $(\theta_2, \phi_2) = (-40°, -20°)$, SNR = 0 dB, and 100 snapshots.

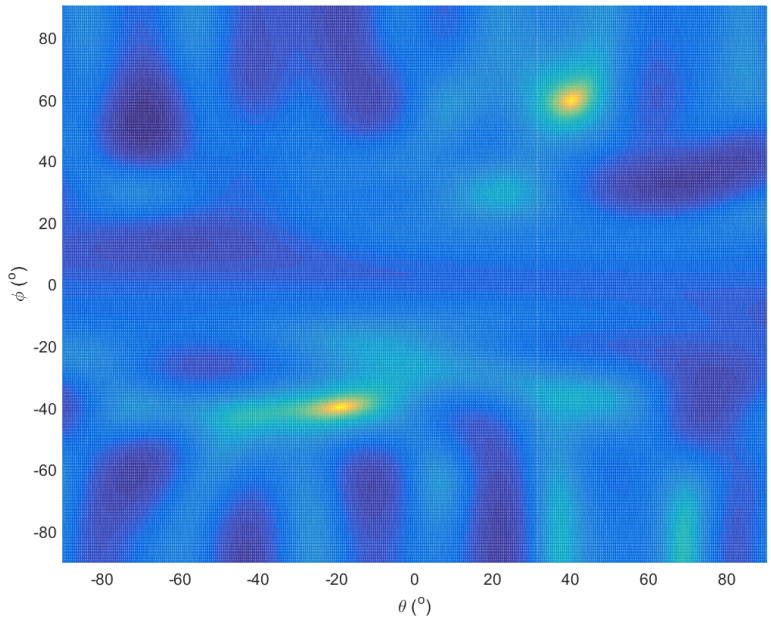

Figure 2.25: Vertical view of the 2-D MUSIC spectrum.

2.3.4 4-D MUSIC for polarized signals

The previous applications of MUSIC algorithm focus on the signals without polarisation. In this part, we also give a review for the MUSIC algorithm applied to the polarized signal scenario. With the two extra polarisation parameters, the traditional MUSIC algorithm can be extended to 4-D parameter estimation. In this thesis, we replace the original omnidirectional sensor by electromagnetic-sensitive sensors [33–35]. A full electromagnetic-sensitive sensor receives the signal in both electric field and magnetic field from x-axis, y-axis and z-axis directions, respectively. In other words, each full electromagnetic-sensitive sensor receives six components of the incoming signals. To simplify the array model, we only keeps two electrical components in x-axis and y-axis direction, and this is called the crossed-dipole sensor array. In the following, most work is based on the crossed-dipole sensor array.

The main problem for 4-D estimation is to estimate the joint steering vector of incoming sources. Generally the model considers the received x-axis and y-axis components in each crossed-dipole sensor as a 2-D vector, and the length of the whole steering vector

becomes $2M$ (M is the length of the sensor array). This is referred to the long-vector model [36–38].

In the long-vector model, with the incoming signal s, the received signals at the $m-th$ senor can be represented as

$$\begin{aligned} r_{xm} &= a_m p_x s \\ r_{ym} &= a_m p_y s \end{aligned} \tag{2.79}$$

where a_m is the phase shift of the source signal at the $m-th$ sensor, and p_x and p_y are the polarisation parameters along the x-axis and y-axis directions, respectively. By combining the two results into 2-D vector, the new result becomes

$$\begin{aligned} \boldsymbol{r}_m &= \begin{bmatrix} r_{xm} \\ r_{ym} \end{bmatrix} = a_m \begin{bmatrix} p_x \\ p_y \end{bmatrix} s \\ &= a_m \cdot \boldsymbol{p} \cdot s \end{aligned} \tag{2.80}$$

where $\boldsymbol{p}$ is the polarisation vector with

$$\boldsymbol{p} = \begin{bmatrix} p_x \\ p_y \end{bmatrix} = \begin{bmatrix} \sin\gamma\cos\theta\cos\phi e^{j\eta} - \cos\gamma\sin\phi \\ \sin\gamma\cos\theta\sin\phi e^{j\eta} + \cos\gamma\cos\phi \end{bmatrix} \tag{2.81}$$

The results for the whole array are denoted by a long vector

$$\begin{aligned} \boldsymbol{r} &= \begin{bmatrix} \boldsymbol{r}_1 \\ \boldsymbol{r}_2 \\ . \\ . \\ . \\ \boldsymbol{r}_M \end{bmatrix} = \begin{bmatrix} a_1\boldsymbol{p}s \\ a_2\boldsymbol{p}s \\ . \\ . \\ . \\ a_M\boldsymbol{p}s \end{bmatrix} \\ &= \boldsymbol{a} \otimes \boldsymbol{p}s \end{aligned} \tag{2.82}$$

where $\boldsymbol{a}$ is the steering vector of the incoming signal and the symbol '$\otimes$' means the Kronecker product. Given two matrices $\boldsymbol{A}, m \times n$ and $\boldsymbol{B}, p \times q$, the Kronecker product is

given as

$$\boldsymbol{A} \otimes \boldsymbol{B} = \begin{bmatrix} a_{11}\boldsymbol{B} & ... & a_{1n}\boldsymbol{B} \\ . & . & . \\ . & . & . \\ . & . & . \\ a_{m1}\boldsymbol{B} & ... & a_{mn}\boldsymbol{B} \end{bmatrix} \tag{2.83}$$

Here we use $\boldsymbol{q}$ to denote the joint steering vector, given by

$$\boldsymbol{q} = \boldsymbol{a} \otimes \boldsymbol{p} \tag{2.84}$$

The joint steering vector $\boldsymbol{q}$ is associated with four parameters $\theta, \phi, \gamma, \eta$. Then the cost function of the 4-D MUSIC estimator can be expressed as

$$P_{MU}(\theta, \phi, \gamma, \eta) = \frac{1}{\boldsymbol{q}^H \boldsymbol{U}_N \boldsymbol{U}_N^H \boldsymbol{q}} \tag{2.85}$$

2.4 Summary

In this chapter a review of polarisation, narrowband beamforming and DOA estimation has been presented. The crossed-dipole sensor array structure is introduced for receiving polarized signals. Capon beamformer and its worst-case constraint application are reviewed for narrowband beamforming. The beampattern and the output SINR curve are given to demonstrate the performance of the algorithm. The MUSIC algorithm is reviewed for DOA estimation, and the algorithm can be extended into multiple-dimensional search. The search spectrum is given for both 1-D and 2-D cases.

Chapter 3

Fully Quaternion-Valued Adaptive Beamforming Based on Crossed-Dipole Arrays

3.1 Introduction

Electromagnetic (EM) vector sensor arrays can track the DOA of impinging signals as well as their polarisation. A crossed-dipole sensor array, firstly introduced in [16] for adaptive beamforming, works by processing the received signals with a long polarisation vector. Based on such a model, the beamforming problem was studied in detail in terms of output signal-to-interference-plus-noise ratio (SINR) [39]. In [13, 40], further detailed analysis was performed showing that the output SINR is affected by DOA and polarisation differences.

Since there are four components for each vector sensor output in a crossed-dipole array, a quaternion model instead of long vectors has been adopted in the past for both adaptive beamforming and direction of arrival estimation [41–48]. In [49], the well-known Capon beamformer was extended to the quaternion domain and a quaternion-valued Capon (Q-Capon) beamformer was proposed with the corresponding optimum solution derived.

However, in most of the beamforming studies, the signal of interest (SOI) is still complex-valued, i.e. with only two components: in-phase (I) and quadrature (Q). Since the output of a quaternion-valued beamformer is also quaternion-valued, only two components of the quaternion are used to recover the SOI, which leads to redundancy in both calculation and data storage. However, with the development of quaternion-valued communications [50, 51], it is very likely that in the future we will have quaternion-valued signals as the SOI, where two traditional complex-valued signals with different polarisations arrive at the antenna array with the same DOA. In such a case, a full quaternion-valued array model is needed to compactly represent the four-component desired signal and also make sure the four components of the quaternion-valued output of the beamformer are fully utilised. In this thesis, we develop such a model and propose a new quaternion-valued Capon beamformer, where both its input and output are quaternion-valued.

Based on the proposed full quaternion model, we further study the robust adaptive beamforming problem in the presence of steering vector errors. In the past, many methods have been proposed to improve the robustness of an adaptive beamformer, such as diagonal loading [6, 52] and those based on the optimization of worst-case performance [53, 54]. In [47], the worst-case based method is extended to the quaternion-valued case for crossed-dipole arrays; however, it is not a full quaternion model since the desired signal is still complex-valued. In this thesis, we extend the worst-case optimisation approach to the full quaternion model and a worst-case based full-quaternion robust adaptive beamforming method is proposed.

3.2 Quaternions

3.2.1 Basics of Quaternion

Quaternion was introduced by W. R. Hamilton [55]. A quaternion is constructed by four components, with one real part and three imaginary parts, defined as

$$q = q_a + iq_b + jq_c + kq_d \tag{3.1}$$

where i, j, k are three basic imaginary units. The multiplication principle among such units is

$$i^2 = j^2 = k^2 = ijk = -1, \tag{3.2}$$

and

$$ij = -ji = k, ki = -ik = j, jk = -kj = i \tag{3.3}$$

The conjugate q^* of q is $q^* = q_a - iq_b - jq_c - kq_d$.

Note that generally the multiplication of quaternions is non-commutative: for $q_1, q_2 \in \mathbb{H}$, $q_1 q_2 \neq q_2 q_1$, where $\mathbb{H}$ represents the quaternion domain. However, if one of the factors, say q_0, is real, then we have $q_0 q_1 = q_1 q_0$.

3.2.2 Quaternion vector and matrix

Quaternion vectors and matrices are quite different from real and complex ones. For quaternion vectors, the space can be denoted by two different bases, left-spanned and right-spanned, which are given by

$$span_L\{u_1, u_2, ...u_M\}, \quad span_R\{u_1, u_2, ...u_M\} \ . \tag{3.4}$$

Here we use $\boldsymbol{q}_L$ to denote the left linear combination case and $\boldsymbol{q}_R$ to denote the right linear combination case. Note that $\boldsymbol{q} = \boldsymbol{q}_L = \boldsymbol{q}_R$,

$$\begin{aligned} \boldsymbol{q}_L &= \sum_{m=1}^{M} \tau_m \mathbf{u}_m \\ \boldsymbol{q}_R &= \sum_{m=1}^{M} \mathbf{u}_m \gamma_m \end{aligned} \tag{3.5}$$

where τ_m,$\gamma_m \in \mathbb{H}$, for $m = 1, 2, ..., M$.

Similarly, the eigendecomposition operation for quaternion matrices also differs from real and complex ones. We can also define the left eigendecomposition $\boldsymbol{Q}_L$ and right eigendecomposition $\boldsymbol{Q}_R$ for a Hermitian matrix $\boldsymbol{Q} \in \mathbb{H}^{N \times N}$, ($\boldsymbol{Q} = \boldsymbol{Q}^H$, where $\{\}^H$ is the Hermitian transpose, a combination of the quaternion-valued conjugate and transpose

operations)

$$\boldsymbol{Q}_L = \sum_{n=1}^{N} \alpha_n \boldsymbol{u}_n \boldsymbol{u}_n^H$$
$$\boldsymbol{Q}_R = \sum_{n=1}^{N} \boldsymbol{u}_n \boldsymbol{u}_n^H \beta_n \tag{3.6}$$

where α and β are the left and right eigenvalues [41, 56, 57]. In this thesis, only right-spanned spaces and left eigendecomposition are employed.

3.2.3 The gradient for a quaternion function

The gradient describes the changing rate of a function at the given point with a certain direction. Unlike the ordinary derivative of a single variable, the gradient of a function contains several multiple variables and appears in a vector. Let $l(\boldsymbol{w})$ be a scalar function of a variable vector $\boldsymbol{w}$, where

$$\boldsymbol{w} = [w_1, w_2, ..., w_N] \tag{3.7}$$

The gradient vector with respect to $\boldsymbol{w}$ is defined by [58]

$$\nabla_{\boldsymbol{w}} l = [\frac{\partial l}{\partial w_1}, \frac{\partial l}{\partial w_2}, ..., \frac{\partial l}{\partial w_N}] \tag{3.8}$$

where '∂' means the partial derivative. When $\boldsymbol{w}$ is real-valued, the elements in the gradient vector, for example $\frac{\partial l}{\partial w_1}$, can be calculated easily. However, when $\boldsymbol{w}$ is complex-valued and even quaternion-valued, it's not straightforward to obtain the results. In the following, firstly a derivation of the partial derivative with respect to a complex-valued variable is given. After that, the results are extended to quaternion field as well.

Consider all variables in $\boldsymbol{w}$ are complex-valued. The first variable w_1 and its conjugate w_1^* are given as

$$w_1 = x + iy, w_1^* = x - iy \tag{3.9}$$

where x, y are the real and imaginary parts of w_1 respectively. Notice that x and y are both real-valued variables and

$$x = \frac{w_1 + w_1^*}{2}, y = \frac{w_1 - w_1^*}{2i} \tag{3.10}$$

For convenience, we ignore the variable $w_2, w_3, ..., w_N$ because they have no influence on computing the partial derivative $\frac{\partial l}{\partial w_1}$. Then the function $l(\boldsymbol{w})$ can be viewed as a function of x and y, which is denoted by $f(x, y)$. As shown in (3.10), the function can be also viewed as a function of w_1 and w_1^*, which can be denoted by $g(w_1, w_1^*)$. Then we have

$$l(\boldsymbol{w}) = f(x, y) = g(w_1, w_1^*) \tag{3.11}$$

According to the chain rule, the partial derivative with respect to x and y can be obtained by

$$\begin{cases} \frac{\partial f}{\partial x} = \frac{\partial g}{\partial w_1}\frac{\partial w_1}{\partial x} + \frac{\partial g}{\partial w_1^*}\frac{\partial w_1^*}{\partial x} \\ \frac{\partial f}{\partial y} = \frac{\partial g}{\partial w_1}\frac{\partial w_1}{\partial y} + \frac{\partial g}{\partial w_1^*}\frac{\partial w_1^*}{\partial y} \end{cases} \tag{3.12}$$

By (3.9), it's clear to know that

$$\begin{cases} \frac{\partial w_1}{\partial x} = 1 \\ \frac{\partial w_1^*}{\partial x} = 1 \\ \frac{\partial w_1}{\partial y} = i \\ \frac{\partial w_1^*}{\partial y} = -i \end{cases} \tag{3.13}$$

Substitute (3.13) into (3.12), we have

$$\begin{cases} \frac{\partial g}{\partial w_1} = \frac{1}{2}(\frac{\partial f}{\partial x} - i\frac{\partial f}{\partial y}) \\ \frac{\partial g}{\partial w_1^*} = \frac{1}{2}(\frac{\partial f}{\partial x} + i\frac{\partial f}{\partial y}) \end{cases} \tag{3.14}$$

Since $l(\boldsymbol{w}) = g(w_1, w_1^*)$, the partial derivative with respect to a complex-valued variable can be denoted as the linear combination of two real-valued variables, where

$$\begin{cases} \frac{\partial l}{\partial w_1} = \frac{1}{2}(\frac{\partial f}{\partial x} - i\frac{\partial f}{\partial y}) \\ \frac{\partial l}{\partial w_1^*} = \frac{1}{2}(\frac{\partial f}{\partial x} + i\frac{\partial f}{\partial y}) \end{cases} \tag{3.15}$$

Specially, when $l(\boldsymbol{w}) = w_1$, we have

$$\begin{cases} \frac{\partial l}{\partial w_1} = \frac{1}{2}(1 + 1) = 1 \\ \frac{\partial l}{\partial w_1^*} = \frac{1}{2}(1 - 1) = 0 \end{cases} \tag{3.16}$$

If $\boldsymbol{w}$ is quaternion-valued, similarly, the first variable w_1 can be extended into four components

$$w_1 = w_{1a} + iw_{1b} + jw_{1c} + kw_{1d} \tag{3.17}$$

where w_{1a}, w_{1b}, w_{1c} and w_{1d} denote the real, i, j and k component of w_1 respectively. These components are all real-valued. Unlike complex-valued variable, these real-valued components cannot be simply denoted as the linear combination of w_1 and its conjugate. However, these components can be connected in another way. Notice that

$$\begin{cases} iw_1i = -w_{1a} - iw_{1b} + jw_{1c} + kw_{1d} \\ jw_1j = -w_{1a} + iw_{1b} - jw_{1c} + kw_{1d} \\ kw_1k = -w_{1a} + iw_{1b} + jw_{1c} - kw_{1d} \end{cases} \tag{3.18}$$

Then it can be found that

$$\begin{cases} w_{1a} = \frac{1}{4}(w_1 - iw_1i - jw_1j - kw_1k) \\ w_{1b} = \frac{1}{4i}(w_1 - iw_1i + jw_1j + kw_1k) \\ w_{1c} = \frac{1}{4j}(w_1 + iw_1i - jw_1j + kw_1k) \\ w_{1d} = \frac{1}{4k}(w_1 + iw_1i + jw_1j - kw_1k) \end{cases} \tag{3.19}$$

Similarly, we ignore other quaternion-valued variables $w_2, w_3, ..., w_N$. The function $l(\boldsymbol{w})$ with respect to quaternion-valued variable w_1 can be viewed as a function $f(w_{1a}, w_{1b}, w_{1c}, w_{1d})$ with respect to real-valued variables w_{1a}, w_{1b}, w_{1c} and w_{1d}. Also, it can be viewed as a function $g(w_1, iw_1i, jw_1j, kw_1k)$ with respect to w_1, iw_1i, jw_1j and kw_1k.

By chain rule, the partial derivatives with respect to w_{1a}, w_{1b}, w_{1c} and w_{1d} can be denoted as

$$\begin{cases} \frac{\partial f}{\partial w_{1a}} = \frac{\partial g}{\partial w_1}\frac{\partial w_1}{\partial w_{1a}} + \frac{\partial g}{\partial iw_1i}\frac{\partial iw_1i}{\partial w_{1a}} + \frac{\partial g}{\partial jw_1j}\frac{\partial jw_1j}{\partial w_{1a}} + \frac{\partial g}{\partial kw_1k}\frac{\partial kw_1k}{\partial w_{1a}} \\ \frac{\partial f}{\partial w_{1b}} = \frac{\partial g}{\partial w_1}\frac{\partial w_1}{\partial w_{1b}} + \frac{\partial g}{\partial iw_1i}\frac{\partial iw_1i}{\partial w_{1b}} + \frac{\partial g}{\partial jw_1j}\frac{\partial jw_1j}{\partial w_{1b}} + \frac{\partial g}{\partial kw_1k}\frac{\partial kw_1k}{\partial w_{1b}} \\ \frac{\partial f}{\partial w_{1c}} = \frac{\partial g}{\partial w_1}\frac{\partial w_1}{\partial w_{1c}} + \frac{\partial g}{\partial iw_1i}\frac{\partial iw_1i}{\partial w_{1c}} + \frac{\partial g}{\partial jw_1j}\frac{\partial jw_1j}{\partial w_{1c}} + \frac{\partial g}{\partial kw_1k}\frac{\partial kw_1k}{\partial w_{1c}} \\ \frac{\partial f}{\partial w_{1d}} = \frac{\partial g}{\partial w_1}\frac{\partial w_1}{\partial w_{1d}} + \frac{\partial g}{\partial iw_1i}\frac{\partial iw_1i}{\partial w_{1d}} + \frac{\partial g}{\partial jw_1j}\frac{\partial jw_1j}{\partial w_{1d}} + \frac{\partial g}{\partial kw_1k}\frac{\partial kw_1k}{\partial w_{1d}} \end{cases} \tag{3.20}$$

Some partial derivatives can be obtained from (3.18), which can be denoted in a matrix

$$\begin{bmatrix} \frac{\partial w_1}{\partial w_{1a}} & \frac{\partial i w_1 i}{\partial w_{1a}} & \frac{\partial j w_1 j}{\partial w_{1a}} & \frac{\partial k w_1 k}{\partial w_{1a}} \\ \frac{\partial w_1}{\partial w_{1b}} & \frac{\partial i w_1 i}{\partial w_{1b}} & \frac{\partial j w_1 j}{\partial w_{1b}} & \frac{\partial k w_1 k}{\partial w_{1b}} \\ \frac{\partial w_1}{\partial w_{1c}} & \frac{\partial i w_1 i}{\partial w_{1c}} & \frac{\partial j w_1 j}{\partial w_{1c}} & \frac{\partial k w_1 k}{\partial w_{1c}} \\ \frac{\partial w_1}{\partial w_{1d}} & \frac{\partial i w_1 i}{\partial w_{1d}} & \frac{\partial j w_1 j}{\partial w_{1d}} & \frac{\partial k w_1 k}{\partial w_{1d}} \end{bmatrix} = \begin{bmatrix} 1 & -1 & -1 & -1 \\ i & -i & i & i \\ j & j & -j & j \\ k & k & k & -k \end{bmatrix} \tag{3.21}$$

After solving (3.20), the partial derivative with respect to the quaternion-valued variable w_1 can be denoted as

$$\frac{\partial l}{\partial w_1} = \frac{1}{4}\left(\frac{\partial f}{\partial w_{1a}} - \frac{\partial f}{\partial w_{1b} i} - \frac{\partial f}{\partial w_{1c} j} - \frac{\partial f}{\partial w_{1d} k}\right) \tag{3.22}$$

In addition, the function $l(\boldsymbol{w})$ can be also viewed as a function of w_1^*, iw_1^*i, jw_1^*j and kw_1^*k. Follow the same derivation, we can also find out the partial derivative with respect to w_1^*, where

$$\frac{\partial l}{\partial w_1^*} = \frac{1}{4}\left(\frac{\partial f}{\partial w_{1a}} + \frac{\partial f}{\partial w_{1b} i} + \frac{\partial f}{\partial w_{1c} j} + \frac{\partial f}{\partial w_{1d} k}\right) \tag{3.23}$$

Specially, when $l(\boldsymbol{w}) = w_1$, we have

$$\begin{cases} \frac{\partial l}{\partial w_1} = \frac{1}{4}(1+1+1+1) = 1 \\ \frac{\partial l}{\partial w_1^*} = \frac{1}{4}(1-1-1-1) = -\frac{1}{2} \end{cases} \tag{3.24}$$

3.3 Model for Crossed-Dipole Arrays

A quaternion number $q = q_a + iq_b + jq_c + kq_d$ can be conveniently expressed as a combination of two complex numbers $c_1 = q_a + jq_c$ and $c_2 = q_b + jq_d$ as follows

$$q = c_1 + ic_2 = (q_a + jq_c) + i(q_b + jq_d) = q_a + iq_b + jq_c + kq_d \tag{3.25}$$

We will use this form later to represent our quaternion-valued signal of interest.

Consider a uniform linear array with N crossed-dipole sensors, as shown in Fig. 5.1, where the adjacent vector sensor spacing d equals half wavelength, and the two components of each crossed-dipole are parallel to $x-$ and $y-$axes, respectively. A quaternion-valued narrowband signal $s_0(t)$ impinges upon the vector sensor array among other M uncorrelated quaternion-valued interfering signals $\{s_m(t)\}_{m=1}^{M}$, with background noise $n(t)$.

$s_0(t)$ can be decomposed into

$$s_0(t) = s_{01}(t) + is_{02}(t) \tag{3.26}$$

where $s_{01}(t)$ and $s_{02}(t)$ are two complex-valued sub-signals with the same DOA but different polarisations.

Assume that all signals are ellipse-polarized. Here we use θ to denote the azimuth angle, ϕ to denote the elevation angle, and γ and η the polarisation parameters of the signal. The parameters, including DOA and polarisation of the m-th signal are denoted by $(\theta_m, \phi_m, \gamma_{m1}, \eta_{m1})$ for the first sub-signal and $(\theta_m, \phi_m, \gamma_{m2}, \eta_{m2})$ for the second sub-signal. Each crossed-dipole sensor receives signals both in the x and y sub-arrays.

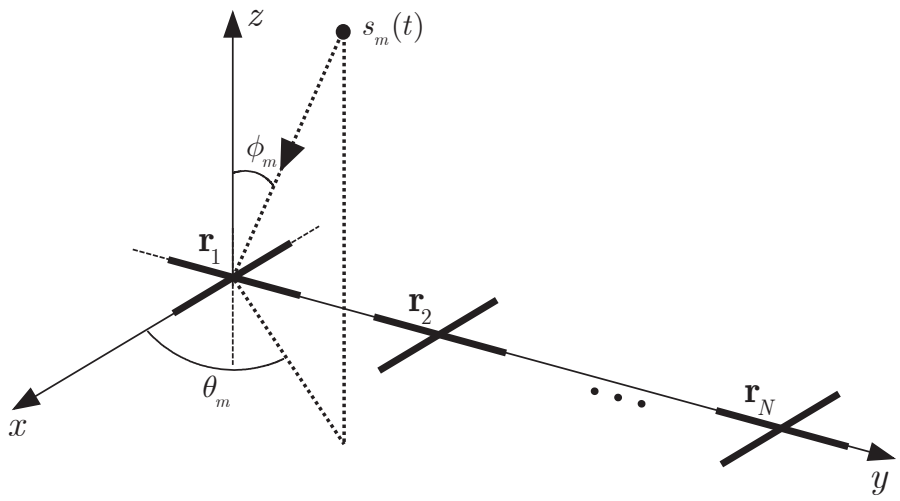

Figure 3.1: A crossed-dipole linear array with N vector sensors.

For signal $s_m(t)$, the corresponding received signals in the x and y sub-arrays are respectively given by [47]:

$$\begin{aligned} \boldsymbol{x}(t) &= \boldsymbol{a}_{m1} p_{xm1} s_{m1}(t) + \boldsymbol{a}_{m2} p_{xm2} s_{m2}(t) \\ \boldsymbol{y}(t) &= \boldsymbol{a}_{m1} p_{ym1} s_{m1}(t) + \boldsymbol{a}_{m2} p_{ym2} s_{m2}(t) \end{aligned} \tag{3.27}$$

where $\boldsymbol{x}(t)$ represents the received part in the x-sub-array and $\boldsymbol{y}(t)$ represents the part in the y-sub-array, and (p_{xm1}, p_{ym1}) and (p_{xm2}, p_{ym2}) are the polarisations of the two complex

sub-signals in x and y directions, respectively, given by,

$$\begin{aligned} p_{xm1} &= -\cos\gamma_{m1} \\ p_{ym1} &= \cos\phi_m \sin\gamma_{m1} e^{j\eta_{m1}} \\ p_{xm2} &= -\cos\gamma_{m2} \\ p_{ym2} &= \cos\phi_m \sin\gamma_{m2} e^{j\eta_{m2}}, \quad \text{when} \quad \theta_m = \frac{\pi}{2} \end{aligned} \tag{3.28}$$

Note that $\boldsymbol{a}_{m1}$ and $\boldsymbol{a}_{m2}$ are the steering vectors for both sub-signals, which are the same since the two sub-signals share the same DOA (θ_m, ϕ_m).

$$\begin{aligned} \boldsymbol{a}_{m1} &= [1, e^{\frac{-j2\pi \sin\theta_m \sin\phi_m}{\lambda}}, ..., e^{\frac{-j(N-1)2\pi \sin\theta_m \sin\phi_m}{\lambda}}]^{\mathrm{T}} \\ \boldsymbol{a}_{m2} &= [1, e^{\frac{-j2\pi \sin\theta_m \sin\phi_m}{\lambda}}, ..., e^{\frac{-j(N-1)2\pi \sin\theta_m \sin\phi_m}{\lambda}}]^{\mathrm{T}} \end{aligned} \tag{3.29}$$

A quaternion model can be constructed by combining the two parts as below:

$$\boldsymbol{q}_m(t) = \boldsymbol{x}(t) + i\boldsymbol{y}(t) \tag{3.30}$$

$$\begin{aligned} &= \boldsymbol{a}_{m1}(p_{xm1} + ip_{ym1})s_{m1}(t) \\ &+ \boldsymbol{a}_{m2}(p_{xm2} + ip_{ym2})s_{m2}(t) \\ &= \boldsymbol{b}_{m1}s_{m1}(t) + \boldsymbol{b}_{m2}s_{m2}(t) \end{aligned} \tag{3.31}$$

where $\{\boldsymbol{b}_{m1}, \boldsymbol{b}_{m2}\} \in \mathbb{H}^{N\times 1}$ can be considered as the composite quaternion-valued steering vector. Combining all source signals and the noise together, the result is given by:

$$\boldsymbol{q}(t) = \sum_{m=0}^{M} (\boldsymbol{b}_{m1}s_{m1}(t) + \boldsymbol{b}_{m2}s_{m2}(t)) + \boldsymbol{n}_q(t) \tag{3.32}$$

where $\boldsymbol{n}_q(t) = \boldsymbol{n}_x(t) + i\boldsymbol{n}_y(t)$ is the quaternion-valued noise vector consisting of the two sub-array noise vectors $\boldsymbol{n}_x(t)$ and $\boldsymbol{n}_y(t)$.

3.4 The Full Quaternion-Valued Capon Beamformer

3.4.1 General algorithm

To recover the SOI among interfering signals and noise, the basic idea is to keep a unity response to the SOI at the beamformer output and then reduce the power variance of the

output as much as possible [18, 59]. The key to construct such a Capon beamformer in the quaternion domain is to design an appropriate constraint to make sure the quaternion-valued SOI can pass through the beamformer with the desired unity response.

Again note that the quaternion-valued SOI can be expressed as a combination of two complex sub-signals. To construct such a constraint, one choice is to make sure the first complex sub-signal of the SOI passes through the beamformer and appear in the real and j components of the beamformer output, while the second complex sub-signal appear in the i and k components of the beamformer output. Then, with a quaternion-valued weight vector $\boldsymbol{w}$, the constraint can be formulated as

$$\boldsymbol{w}^H \boldsymbol{C} = \boldsymbol{f} \tag{3.33}$$

where $\boldsymbol{C} = [\boldsymbol{b}_{01} \quad \boldsymbol{b}_{02}]$, $\boldsymbol{f} = [1 \quad i]$.

With this constraint, the beamformer output $z(t)$ is given by

$$\begin{aligned} z(t) &= \boldsymbol{w}^H \boldsymbol{q}(t) \\ &= \underbrace{s_{01}(t) + i s_{02}(t)}_{s_0(t)} + \boldsymbol{w}^H \boldsymbol{n}_q(t) \\ &\quad + \sum_{m=1}^{M} \boldsymbol{w}^H [\boldsymbol{b}_{m1} s_{m1}(t) + \boldsymbol{b}_{m2} s_{m2}(t)] \end{aligned} \tag{3.34}$$

Clearly, the quaternion-valued SOI has been preserved at the output with the desired unity response.

Now, the full-quaternion Capon (full Q-Capon) beamformer can be formulated as

$$\begin{aligned} \min \quad & \boldsymbol{w}^H \boldsymbol{R} \boldsymbol{w} \\ \text{s.t.} \quad & \boldsymbol{w}^H \boldsymbol{C} = \boldsymbol{f} \end{aligned} \tag{3.35}$$

where

$$\boldsymbol{R} = E\{\boldsymbol{q}(t) \boldsymbol{q}^H(t)\} \tag{3.36}$$

Applying the Lagrange multipliers method, we have

$$l(\boldsymbol{w}, \boldsymbol{\lambda}) = \boldsymbol{w}^H \boldsymbol{R} \boldsymbol{w} + (\boldsymbol{w}^H \boldsymbol{C} - \boldsymbol{f}) \boldsymbol{\lambda}^H + \boldsymbol{\lambda} (\boldsymbol{C}^H \boldsymbol{w} - \boldsymbol{f}^H) \tag{3.37}$$

where $\boldsymbol{\lambda}$ is a quaternion-valued vector of the Lagrange multipliers.

The minimum can be obtained by setting the gradient of (3.37) with respect to $\boldsymbol{w}^*$ equal to a zero vector [58]. It is given by

$$\nabla_{\boldsymbol{w}^*} l(\boldsymbol{w}, \boldsymbol{\lambda}) = \frac{1}{2}\boldsymbol{R}\boldsymbol{w} + \frac{1}{2}\boldsymbol{C}\boldsymbol{\lambda}^H = \boldsymbol{0} \tag{3.38}$$

Considering all the constraints above, we obtain the optimum weight vector $\boldsymbol{w}_{opt}$ as follows

$$\boldsymbol{w}_{opt} = \boldsymbol{R}^{-1}\boldsymbol{C}(\boldsymbol{C}^H\boldsymbol{R}^{-1}\boldsymbol{C})^{-1}\boldsymbol{f}^H \tag{3.39}$$

A detailed derivation for the quaternion-valued optimum weight vector can be found at later subsection.

In the next, we give a brief analysis to show that by this optimum weight vector, the interference part at the beamformer output $z(t)$ in (3.34) has been suppressed effectively.

Expanding the covariance matrix, we have

$$\boldsymbol{R} = E\{\boldsymbol{q}(t)\boldsymbol{q}^H(t)\} = \boldsymbol{R}_{i+n} + \sigma_1^2\boldsymbol{b}_{01}\boldsymbol{b}_{01}^H + \sigma_2^2\boldsymbol{b}_{02}\boldsymbol{b}_{02}^H \tag{3.40}$$

where σ_1^2, σ_2^2 are the power of the two sub-signals of SOI and $\boldsymbol{R}_{i+n}$ denotes the covariance matrix of interferences plus noise. Using (3.39), we have

$$\boldsymbol{w}_{opt} = \boldsymbol{R}_{i+n}^{-1}\boldsymbol{C}\boldsymbol{\beta} \tag{3.41}$$

where $\boldsymbol{\beta} = (\boldsymbol{C}^H\boldsymbol{R}_{i+n}\boldsymbol{C})^{-1}\boldsymbol{f}^H \in \mathbb{H}^{2\times 1}$ is a quaternion vector.

Applying left eigendecomposition for quaternion matrices,

$$\boldsymbol{R}_{i+n} = \sum_{n=1}^{N} \alpha_n \boldsymbol{u}_n \boldsymbol{u}_n^H \tag{3.42}$$

with $\alpha_1 \geq ... \geq \alpha_{M-2} > \alpha_{M-1} = ... = \alpha_N = 2\sigma_0^2 \in \mathbb{R}$, where $2\sigma_0^2$ denotes the noise power.

With sufficiently high interference to noise ratio (INR), the inverse of $\boldsymbol{R}_{i+n}$ can be approximated by

$$\boldsymbol{R}_{i+n}^{-1} \approx \sum_{n=M+1}^{N} \frac{1}{2\sigma_0^2}\boldsymbol{u}_n \boldsymbol{u}_n^H \tag{3.43}$$

Then we have

$$\boldsymbol{w}_{opt} = \sum_{n=M+1}^{N} \frac{1}{2\sigma_0^2} \boldsymbol{u}_n \boldsymbol{u}_n^H \boldsymbol{C}\boldsymbol{\beta} = \sum_{n=M+1}^{N} \boldsymbol{u}_n \rho_n \tag{3.44}$$

where $\rho_n = \frac{1}{2\sigma_0^2} \boldsymbol{u}_n^H \boldsymbol{C}\boldsymbol{\beta}$. Since $\boldsymbol{u}_n^H \in \mathbb{H}^{1\times N}$, $\boldsymbol{C} \in \mathbb{H}^{N\times 2}$ and $\boldsymbol{\beta} \in \mathbb{H}^{2\times 1}$, ρ_n is a quaternion-valued scalar. Clearly, $\boldsymbol{w}_{opt}$ is the right linear combination of $\{u_{M+1}, u_{M+2}, ..., u_N\}$, and $\boldsymbol{w} \in span_R\{u_{M+1}, u_{M+2}, ..., u_N\}$.

For those M interfering signals, their quaternion steering vectors belong to the space right-spanned by the related M eigenvectors, i.e. $\boldsymbol{b}_{m1}, \boldsymbol{b}_{m2} \in span_R\{u_1, u_2, ..., u_M\}$. As a result,

$$\boldsymbol{w}_{opt}^H \boldsymbol{b}_{m1} \approx 0, \quad \boldsymbol{w}_{opt}^H \boldsymbol{b}_{m2} \approx 0, \quad m = 1, 2, ..., M \tag{3.45}$$

which shows that the beamformer has eliminated the interferences effectively.

3.4.2 Derivation for the quaternion-valued optimum weight vector

The gradient of a quaternion vector $\mathbf{u} = \boldsymbol{w}^H \boldsymbol{C}\boldsymbol{\lambda}^H$ with respect to $\boldsymbol{w}^*$ can be calculated as follows:

$$\nabla_{\boldsymbol{w}^*}\mathbf{u} = [\nabla_{w_1^*}\mathbf{u} \quad \nabla w_2^*\mathbf{u} \quad ...\nabla_{w_n^*}\mathbf{u}]^T \tag{3.46}$$

where w_n, $n = 1, 2, \cdots, N$ is the n-th quaternion-valued coefficient of the beamformer. Then,

$$\nabla_{w_1^*}\mathbf{u} = \frac{1}{4}(\nabla_{w_{1a}}\mathbf{u} + \nabla_{w_{1b}}\mathbf{u}i + \nabla_{w_{1c}}\mathbf{u}j + \nabla_{w_{1d}}\mathbf{u}k) \tag{3.47}$$

where

$$w_1^* = w_{1a} - w_{1b}i - w_{1c}j - w_{1d}k \tag{3.48}$$

Since w_{1a} is real-valued, with the chain rule [58], we have

$$\begin{aligned} \nabla_{w_{1a}}\mathbf{u} &= \nabla_{w_{1a}}(\boldsymbol{w}^H)\boldsymbol{C}\boldsymbol{\lambda}^H + \boldsymbol{w}^H\nabla_{w_{1a}}(\boldsymbol{C}\boldsymbol{\lambda}^H) \\ &= [1 \quad 0 \quad 0 \quad ... \quad 0]\boldsymbol{C}\boldsymbol{\lambda}^H \end{aligned} \tag{3.49}$$

Similarly,

$$\begin{aligned}\nabla_{w_{1b}}\mathbf{u} &= [-i \quad 0 \quad 0 \quad ... \quad 0]\boldsymbol{C}\boldsymbol{\lambda}^H \\ \nabla_{w_{1c}}\mathbf{u} &= [-j \quad 0 \quad 0 \quad ... \quad 0]\boldsymbol{C}\boldsymbol{\lambda}^H \\ \nabla_{w_{1d}}\mathbf{u} &= [-k \quad 0 \quad 0 \quad ... \quad 0]\boldsymbol{C}\boldsymbol{\lambda}^H \end{aligned} \tag{3.50}$$

Hence,

$$\nabla_{w_1^*}\mathbf{u} = \frac{1}{4}(4Re\{\boldsymbol{C}\boldsymbol{\lambda}^H\}_1) = Re\{\boldsymbol{C}\boldsymbol{\lambda}^H\}_1 \tag{3.51}$$

where the subscript $\{\}_1$ in the last item means taking the first entry of the vector.

Finally,

$$\nabla_{\boldsymbol{w}^*}\mathbf{u} = Re\{\boldsymbol{C}\boldsymbol{\lambda}^H\} \tag{3.52}$$

The gradient of the quaternion vector $\mathbf{v} = \boldsymbol{\lambda}\boldsymbol{C}^H\boldsymbol{w}$ with respect to $\boldsymbol{w}^*$ can be calculated in the same way:

$$\begin{aligned}\nabla_{w_{1a}}\mathbf{v} &= \boldsymbol{\lambda}\boldsymbol{C}^H\nabla_{w_{1a}}\boldsymbol{w} + \nabla_{w_{1a}}(\boldsymbol{\lambda}\boldsymbol{C}^H)\boldsymbol{w} \\ &= \boldsymbol{\lambda}\boldsymbol{C}^H[1 \quad 0 \quad 0 \quad ... \quad 0]^{\mathrm{T}}\end{aligned} \tag{3.53}$$

Similarly,

$$\begin{aligned}\nabla_{w_{1b}}\mathbf{v} &= \boldsymbol{\lambda}\boldsymbol{C}^H[i \quad 0 \quad 0 \quad ... \quad 0]^{\mathrm{T}} \\ \nabla_{w_{1c}}\mathbf{v} &= \boldsymbol{\lambda}\boldsymbol{C}^H[j \quad 0 \quad 0 \quad ... \quad 0]^{\mathrm{T}} \\ \nabla_{w_{1d}}\mathbf{v} &= \boldsymbol{\lambda}\boldsymbol{C}^H[k \quad 0 \quad 0 \quad ... \quad 0]^{\mathrm{T}}\end{aligned} \tag{3.54}$$

Thus, the gradient can be expressed as

$$\nabla_{w_1^*}\mathbf{v} = -\frac{1}{2}\{\boldsymbol{C}\boldsymbol{\lambda}^H\}_1^* \tag{3.55}$$

Finally,

$$\nabla_{\boldsymbol{w}^*}\mathbf{v} = -\frac{1}{2}(\boldsymbol{C}\boldsymbol{\lambda}^H)^* \tag{3.56}$$

The gradient of $c_{\mathbf{w}} = \boldsymbol{w}^H\boldsymbol{R}\boldsymbol{w}$ can be calculated as follows.

$$\nabla_{\boldsymbol{w}^*}c_{\mathbf{w}} = [\nabla_{w_1^*}c_{\mathbf{w}} \quad \nabla_{w_2^*}c_{\mathbf{w}} \quad ... \quad \nabla_{w_n^*}c_{\mathbf{w}}]^{\mathrm{T}} \tag{3.57}$$

$$\nabla_{w_1^*} c_{\mathbf{w}} = \frac{1}{4}(\nabla_{w_{1a}} c_{\mathbf{w}} + \nabla_{w_{1b}} c_{\mathbf{w}} i + \nabla_{w_{1c}} c_{\mathbf{w}} j + \nabla_{w_{1d}} c_{\mathbf{w}} k) \tag{3.58}$$

Now we calculate the gradient of $c_{\mathbf{w}}$ with respect to the four components of w_1.

$$\begin{aligned} \nabla_{w_{1a}} c_{\mathbf{w}} &= \nabla_{w_{1a}} (\boldsymbol{w}^H \boldsymbol{R}) \boldsymbol{w} + \boldsymbol{w}^H \boldsymbol{R} \nabla_{w_{1a}} \boldsymbol{w} \\ &= [1 \quad 0 \quad 0 \quad ... \quad 0] \boldsymbol{R} \boldsymbol{w} \\ &\quad + \boldsymbol{w}^H \boldsymbol{R} [1 \quad 0 \quad 0 \quad ... \quad 0]^{\mathrm{T}} \end{aligned} \tag{3.59}$$

The other three components are,

$$\begin{aligned} \nabla_{w_{1b}} c_{\mathbf{w}} &= [-i \quad 0 \quad 0 \quad ... \quad 0] \boldsymbol{R} \boldsymbol{w} \\ &\quad + \boldsymbol{w}^H \boldsymbol{R} [i \quad 0 \quad 0 \quad ... \quad 0]^{\mathrm{T}} \\ \nabla_{w_{1c}} c_{\mathbf{w}} &= [-j \quad 0 \quad 0 \quad ... \quad 0] \boldsymbol{R} \boldsymbol{w} \\ &\quad + \boldsymbol{w}^H \boldsymbol{R} [j \quad 0 \quad 0 \quad ... \quad 0]^{\mathrm{T}} \\ \nabla_{w_{1d}} c_{\mathbf{w}} &= [-k \quad 0 \quad 0 \quad ... \quad 0] \boldsymbol{R} \boldsymbol{w} \\ &\quad + w^H \boldsymbol{R} [k \quad 0 \quad 0 \quad ... \quad 0]^{\mathrm{T}} \end{aligned}$$

Hence,

$$\nabla_{w_1^*} c_{\mathbf{w}} = Re\{\boldsymbol{R}\boldsymbol{w}\}_1 - \frac{1}{2}\{\boldsymbol{R}\boldsymbol{w}\}_1^* = \frac{1}{2}\{\boldsymbol{R}\boldsymbol{w}\}_1 \tag{3.60}$$

Finally,

$$\nabla_{\boldsymbol{w}^*} c_{\mathbf{w}} = \frac{1}{2}\boldsymbol{R}\boldsymbol{w} \tag{3.61}$$

Combining (3.52), (3.56) and (3.61), with (3.37), we have

$$\nabla_{\boldsymbol{w}^*} l(\boldsymbol{w}, \boldsymbol{\lambda}) = \frac{1}{2}(\boldsymbol{R}\boldsymbol{w} + \boldsymbol{C}\boldsymbol{\lambda}^H) = \mathbf{0} \tag{3.62}$$

Further,

$$\boldsymbol{w} = -\boldsymbol{R}^{-1}\boldsymbol{C}\boldsymbol{\lambda}^H \tag{3.63}$$

Subsituting (3.63) into (3.33),

$$\boldsymbol{\lambda} = -\boldsymbol{f}(\boldsymbol{C}^H \boldsymbol{R}^{-1} \boldsymbol{C})^{-1} \tag{3.64}$$

Finally,

$$\boldsymbol{w} = \boldsymbol{R}^{-1}\boldsymbol{C}(\boldsymbol{C}^H \boldsymbol{R}^{-1} \boldsymbol{C})^{-1}\boldsymbol{f}^H \tag{3.65}$$

3.4.3 Complexity analysis

In this section, we make a comparison of the computation complexity between the Q-Capon beamformer in [49] and our proposed full Q-Capon beamformer. To deal with a quaternion-valued signal, the Q-Capon beamformer has to process the two complex sub-signals separately to recover the desired signal completely, which means we need to apply the beamformer twice for a quaternion-valued SOI. However, for the full Q-Capon beamformer, it is not needed and the SOI is recovered directly by applying the beamformer once.

For the Q-Capon beamformer, the weight vector is calculated by

$$\boldsymbol{w} = \boldsymbol{R}^{-1}\boldsymbol{a}_0(\boldsymbol{a}_0^H\boldsymbol{R}^{-1}\boldsymbol{a}_0)^{-1} \tag{3.66}$$

where $\mathbf{a}_0$ is the steering vector for the complex-valued SOI. To calculate the weight vector, firstly we need to evaluate the computation complexity of the matrix inversion. In the following, we use Gaussian elimination to calculate the matrix inversion. The Gaussian elimination can be concluded as several steps:

- 1. Expand the original matrix with a identity matrix of the same dimension.
- 2. Transform the original matrix to a upper triangular matrix by elementary row operations.
- 3. By dividing the right coefficients to make every diagonal entry to be 1.
- 4. Continue to transform the upper triangular matrix to a diagonal matrix.
- 5. The identity matrix follows the same row operations in steps 2,3 and 4. The resultant matrix is the inverse of the original matrix.

Here we have assumed that the covariance matrix is acquired in advance and the array have N crossed-dipole sensors. To calculate the inversion of the given matrix $\boldsymbol{R}$, we can follow the steps 1, 2 and 3 first. To transform the matrix to a triangular matrix and make every diagonal entry to be 1, we need to transform the lower elements to zeros of each

column. For the first column, the zero entries can be approached by $N-1$ multiplications, $N-1$ additions and N divisions. As the operation is the row operation, the number of multiplications and additions should be multiplied by the length of the row vector N. However there is no increase of the divisions since the division operation is calculated based on row vectors. In total, the first column needs $N(N-1)$ multiplication-additions and N divisions. In the second column, since there are $N-1$ zero entries in the first column, the operations will be reduced to $N-2$ multiplication-additions and $N-1$ divisions. As the new row vectors have a zero entry, the length is also reduced to $N-1$. As a result, the operations to calculate the second column include $(N-1)(N-2)$ multiplication-additions and $N-1$ divisions. Similarly, in the third column, $(N-2)(N-3)$ multiplication-additions and $N-2$ divisions are needed. Finally, the multiplication-additions needed to transform an upper triangular matrix is

$$\sum_{n=1}^{N} n(n-1) = \sum_{n=1}^{N} n^2 - \sum_{n=1}^{N} n = \frac{N^3 - N}{3} \tag{3.67}$$

And the number of division operations is

$$\sum_{n=1}^{N} n = \frac{N(N+1)}{2} \tag{3.68}$$

Next we transform the matrix to a diagonal matrix. The $N-th$ column needs $N-1$ multiplication-additions to ensure zero entries. However, there is no need for division operations because the last element of this column is 1. Similarly, the $(N-1)-th$ column needs $N-2$ multiplication-additions, and the first column needs 0 operations. Hence, the total number of multiplication-addition operations needed for the diagonal matrix is

$$\sum_{n=1}^{N} (n-1) = \frac{N(N-1)}{2} \tag{3.69}$$

The total number of multiplication-addition operations for the whole matrix inversion is

$$\frac{N^3 - N}{3} + \frac{N(N-1)}{2} = \frac{2N^3 + 3N^2 - 5N}{6} \tag{3.70}$$

The total operations for matrix inversion include $\frac{2N^3+3N^2-5N}{6}$ multiplications, $\frac{2N^3+3N^2-5N}{6}$ additions and $\frac{N(N+1)}{2}$ divisions.

Since $\boldsymbol{R}^{-1}$ is a quaternion-valued matrix, all the operations are based on quaternion-valued. To calculate the product $\boldsymbol{R}^{-1}\boldsymbol{a}_0$, N^2 multiplication and $N(N-1)$ addition operations are needed. There are extra N multiplications and $N-1$ additions in the whole product $\boldsymbol{a}_0^H\boldsymbol{R}^{-1}\boldsymbol{a}_0$. As the product is a scalar, the inversion only needs 1 division operation. Based on the operations, calculating the result of $(\boldsymbol{a}_0^H\boldsymbol{R}^{-1}\boldsymbol{a}_0)^{-1}$ needs N^2+N multiplications, N^2-1 additions and 1 division. Finally, the weight vector $\boldsymbol{w}$ is calculated by the product of $\boldsymbol{R}^{-1}\boldsymbol{a}_0$ and $(\boldsymbol{a}_0^H\boldsymbol{R}^{-1}\boldsymbol{a}_0)^{-1}$, which needs N multiplications. In conclusion, the number of quaternion-valued operations to calculate a weight vector is

$$\begin{aligned} &2\cdot\frac{2N^3+3N^2-5N}{6}+\frac{N(N+1)}{2}+N^2+N+N^2-1+1+N \\ &=\frac{2}{3}N^3+\frac{7}{2}N^2+\frac{5}{6}N \end{aligned} \tag{3.71}$$

For a quaternion-valued signal, the Q-Capon beamformer needs two beamforming operations to reconstruct the signal. Then the total number of operations needed is

$$\begin{aligned} &2\cdot(\frac{2}{3}N^3+\frac{7}{2}N^2+\frac{5}{6}N) \\ &=\frac{4}{3}N^3+7N^2+\frac{5}{3}N \end{aligned} \tag{3.72}$$

For the proposed full Q-Capon beamformer, the weight vector is acquired by the product of $\boldsymbol{R}^{-1}\boldsymbol{C}(\boldsymbol{C}^H\boldsymbol{R}^{-1}\boldsymbol{C})^{-1}\boldsymbol{f}^H$. The operation for matrix inversion has been discussed in the Q-Capon beamformer part, and here we first focus on the product $\boldsymbol{R}^{-1}\boldsymbol{C}$. It can be obtained that this product needs $2N^2$ multiplications and $2N(N-1)$ additions. Calculating $\boldsymbol{C}^H\boldsymbol{R}^{-1}\boldsymbol{C}$ needs extra $4N$ multiplications and $4(N-1)$ additions. The total operation number of $\boldsymbol{C}^H\boldsymbol{R}^{-1}\boldsymbol{C}$ is $2N^2+4N$ multiplications and $(2N+4)(N-1)$ additions. To calculate the inverse of the 2×2 matrix, 3 multiplications, 3 additions and 3 divisions are needed. There are $2N$ multiplications and N additions with the product of $\boldsymbol{f}^H$. The total operations needed for the full Q-Capon beamformer

$$\begin{aligned} &2\cdot\frac{2N^3+3N^2-5N}{6}+\frac{N(N+1)}{2}+2N^2+4N+(2N+4)(N-1)+9+3N \\ &=\frac{2}{3}N^3+\frac{11}{2}N^2+\frac{47}{6}N+5 \end{aligned} \tag{3.73}$$

Ignoring the lower order of the results, the Q-Capon beamformer approximately requires $\frac{4}{3}N^3$ operations and the full Q-Capon beamformer requires $\frac{2}{3}N^3$ quaternion-valued operations, which indicates the full Q-Capon beamformer has less computational complexity than the original Q-Capon beamformer. When the sensor number N approaches a large number, the full Q-Capon beamformer takes half the complexity of a Q-Capon beamformer.

3.5 Worst-case Based Robust Adaptive Beamforming

3.5.1 Worst-case constrained algorithm

The proposed full quaternion beamformer is based on accurate steering vectors of incident array signals. However, steering vector mismatch usually exists in array processing as a result of DOA or polarisation mismatch and various model errors. With such a mismatch, there will be an error vector $\boldsymbol{e}$ ($|\boldsymbol{e}| \leq \varepsilon$, where ε is the upper bound of its norm) between the assumed quaternion-valued steering vector $\boldsymbol{b}_m$ and the actual quaternion-valued steering vector $\bar{\boldsymbol{b}}_m$, which can be expressed as [47]

$$\bar{\boldsymbol{b}}_m = \boldsymbol{b}_m + \boldsymbol{e} \tag{3.74}$$

According to (3.74), the actual quaternion-valued steering vector could be any vector within the multi-dimensional sphere, centered at $\boldsymbol{b}_m$ with a radius ε.

In order to achieve a robust response, the beamformer can be constrained to have a response greater than unity for all steering vectors within such a sphere. For the proposed quaternion-valued beamformer in the last section, the constraint can be formulated as

$$\begin{aligned} & \min \boldsymbol{w}^H \boldsymbol{R} \boldsymbol{w} \\ \text{s.t.} \quad & \min |\boldsymbol{w}^H \bar{\boldsymbol{b}}_{01}| \geq 1, \quad \min |\boldsymbol{w}^H \bar{\boldsymbol{b}}_{02}| \geq 1 \end{aligned} \tag{3.75}$$

According to (3.74), for the first constraint (corresponding to $\bar{\boldsymbol{b}}_{01}$), we have

$$|\boldsymbol{w}^H \bar{\boldsymbol{b}}_{01}| = |\boldsymbol{w}^H \boldsymbol{b}_{01} + \boldsymbol{w}^H \boldsymbol{e}| \geq |\boldsymbol{w}^H \boldsymbol{b}_{01}| - |\boldsymbol{w}^H \boldsymbol{e}| \ . \tag{3.76}$$

Since $|\boldsymbol{e}| \leq \varepsilon$, we further have

$$|\boldsymbol{w}^H \bar{\boldsymbol{b}}_{01}| \geq |\boldsymbol{w}^H \boldsymbol{b}_{01}| - |\boldsymbol{w}^H \boldsymbol{e}| \geq |\boldsymbol{w}^H \boldsymbol{b}_{01}| - \varepsilon||\boldsymbol{w}|| \tag{3.77}$$

From (3.77), it can be derived that

$$\begin{aligned} \min |\boldsymbol{w}^H \bar{\boldsymbol{b}}_{01}| &= |\boldsymbol{w}^H \boldsymbol{b}_{01}| - \varepsilon||\boldsymbol{w}|| \geq 1 \\ |\boldsymbol{w}^H \boldsymbol{b}_{01}| &\geq 1 + \varepsilon||\boldsymbol{w}|| \end{aligned} \tag{3.78}$$

Now the constraint becomes

$$\begin{aligned} &\min \boldsymbol{w}^H \boldsymbol{R} \boldsymbol{w} \\ \text{s.t.} \quad &|\boldsymbol{w}^H \boldsymbol{b}_{01}| \geq 1 + \varepsilon||\boldsymbol{w}|| \end{aligned} \tag{3.79}$$

We should ensure that the response for the first sub-signal $s_{01}(t)$ is constrained in the real and j domain and the response for $s_{02}(t)$ in the i and k domain. For a given quaternion-valued vector $\boldsymbol{w}$, the final output power $\boldsymbol{w}^H \boldsymbol{R} \boldsymbol{w}$ keeps unchanged if $\boldsymbol{w}$ undergoes any phase shift. That is to say, we can always find a quaternion-valued vector $\boldsymbol{w}$ to make the product $\boldsymbol{w}^H \boldsymbol{b}_{01}$ real-valued. Then, the constraint can be changed to

$$\begin{aligned} &\min \boldsymbol{w}^H \boldsymbol{R} \boldsymbol{w} \\ \text{s.t.} \quad &Re\{\boldsymbol{w}^H \boldsymbol{b}_{01}\} \geq 1 + \varepsilon||\boldsymbol{w}|| \\ &Im^{(i)}\{\boldsymbol{w}^H \boldsymbol{b}_{01}\} = 0 \\ &Im^{(j)}\{\boldsymbol{w}^H \boldsymbol{b}_{01}\} = 0 \\ &Im^{(k)}\{\boldsymbol{w}^H \boldsymbol{b}_{01}\} = 0 \end{aligned} \tag{3.80}$$

We can transform the constraint of the $\bar{\boldsymbol{b}}_{02}$ part in the same way. Then, the overall

formulation for the worst-case based method can be described as

$$\begin{aligned} & \min \boldsymbol{w}^H \boldsymbol{R} \boldsymbol{w} \\ \text{s.t.} \quad & Re\{\boldsymbol{w}^H \boldsymbol{b}_{01}\} \geq 1 + \varepsilon ||\boldsymbol{w}|| \\ & Im^{(i)}\{\boldsymbol{w}^H \boldsymbol{b}_{01}\} = 0 \\ & Im^{(j)}\{\boldsymbol{w}^H \boldsymbol{b}_{01}\} = 0 \\ & Im^{(k)}\{\boldsymbol{w}^H \boldsymbol{b}_{01}\} = 0 \\ & Re\{\boldsymbol{w}^H \boldsymbol{b}_{02}\} = 0 \\ & Im^{(i)}\{\boldsymbol{w}^H \boldsymbol{b}_{02}\} \geq 1 + \varepsilon ||\boldsymbol{w}|| \\ & Im^{(j)}\{\boldsymbol{w}^H \boldsymbol{b}_{02}\} = 0 \\ & Im^{(k)}\{\boldsymbol{w}^H \boldsymbol{b}_{02}\} = 0 \end{aligned} \tag{3.81}$$

This is a convex optimization problem. A convex optimization problem is a problem where all of the constraints are convex functions, and the objective is a convex function if minimizing, or a concave function if maximizing. We refer the convex optimization implementation as the full quaternion worst-case constraint beamformer (FQWCCB).

3.5.2 Optimization implementation of FQWCCB

The first constraint in (3.81) is to minimize $\boldsymbol{w}^H \boldsymbol{R} \boldsymbol{w}$. To further simplify the constraint, it's reasonable to apply Cholesky decomposition on $\boldsymbol{R}$. The Cholesky decomposition is a decomposition of a Hermitian, positive-definite matrix into the product of a lower triangular matrix and its conjugate transpose. As discussed in former sections, $\boldsymbol{R}$ is a Hermitian matrix with all positive eigenvalues. The Cholesky decomposition of $\boldsymbol{R}$ is in the form [60]

$$\boldsymbol{R} = \boldsymbol{Q} \boldsymbol{Q}^H \tag{3.82}$$

where $\boldsymbol{Q}$ is a lower triangular matrix with real and positive diagonal entries. As $\boldsymbol{Q}^H$ is the conjugate transpose of $\boldsymbol{Q}$, it's a upper triangular matrix with the same diagonal

entries as $\boldsymbol{Q}$. For every Hermitian and positive-definite matrix, there is a unique Cholesky decomposition.

After the decomposition, the output power $\boldsymbol{w}^H \boldsymbol{R} \boldsymbol{w}$ can be rewritten as $\boldsymbol{w}^H \boldsymbol{Q} \boldsymbol{Q}^H \boldsymbol{w}$, which is the l-2 norm of the vector $\boldsymbol{w}^H \boldsymbol{Q}$. To simplify the computation, the constraint can be transformed into

$$\begin{aligned} & \min ||\boldsymbol{w}^H \boldsymbol{Q}|| \\ \text{s.t.} \quad & Re\{\boldsymbol{w}^H \boldsymbol{b}_{01}\} \geq 1 + \varepsilon ||\boldsymbol{w}|| \\ & Im^{(i)}\{\boldsymbol{w}^H \boldsymbol{b}_{01}\} = 0 \\ & Im^{(j)}\{\boldsymbol{w}^H \boldsymbol{b}_{01}\} = 0 \\ & Im^{(k)}\{\boldsymbol{w}^H \boldsymbol{b}_{01}\} = 0 \\ & Re\{\boldsymbol{w}^H \boldsymbol{b}_{02}\} = 0 \\ & Im^{(i)}\{\boldsymbol{w}^H \boldsymbol{b}_{02}\} \geq 1 + \varepsilon ||\boldsymbol{w}|| \\ & Im^{(j)}\{\boldsymbol{w}^H \boldsymbol{b}_{02}\} = 0 \\ & Im^{(k)}\{\boldsymbol{w}^H \boldsymbol{b}_{02}\} = 0 \end{aligned} \tag{3.83}$$

Note that $\boldsymbol{Q}, \boldsymbol{w}, \boldsymbol{b}_{01}$ and $\boldsymbol{b}_{02}$ are quaternion-valued. As we cannot use quaternion-valued variables directly with CVX tools in matlab. As a result, it is necessary to convert the quaternion-valued matrices or vectors into real-valued ones.

We can rewrite $\boldsymbol{Q}, \boldsymbol{w}, \boldsymbol{b}_{01}$ and $\boldsymbol{b}_{02}$ as follows

$$\begin{aligned} \boldsymbol{Q} &= \boldsymbol{Q}_1 + \boldsymbol{Q}_2 i + \boldsymbol{Q}_3 j + \boldsymbol{Q}_4 k \\ \boldsymbol{w} &= \boldsymbol{w}_1 + \boldsymbol{w}_2 i + \boldsymbol{w}_3 j + \boldsymbol{w}_4 k \\ \boldsymbol{b}_{01} &= \boldsymbol{b}_{01,1} + \boldsymbol{b}_{01,2} i + \boldsymbol{b}_{01,3} j + \boldsymbol{b}_{01,4} k \\ \boldsymbol{b}_{02} &= \boldsymbol{b}_{02,1} + \boldsymbol{b}_{02,2} i + \boldsymbol{b}_{02,3} j + \boldsymbol{b}_{02,4} k \end{aligned} \tag{3.84}$$

where $\boldsymbol{Q}_l, \boldsymbol{w}_l, \boldsymbol{b}_{01,l}$ and $\boldsymbol{b}_{02,l} (1 \leq l \leq 4)$ are all real-valued.

We can also define a new form for the quaternion-valued matrices

$$\bar{\boldsymbol{Q}} \triangleq \begin{bmatrix} \boldsymbol{Q}_1 & \boldsymbol{Q}_2 & \boldsymbol{Q}_3 & \boldsymbol{Q}_4 \\ -\boldsymbol{Q}_2 & \boldsymbol{Q}_1 & -\boldsymbol{Q}_4 & \boldsymbol{Q}_3 \\ -\boldsymbol{Q}_3 & \boldsymbol{Q}_4 & \boldsymbol{Q}_1 & -\boldsymbol{Q}_2 \\ -\boldsymbol{Q}_4 & -\boldsymbol{Q}_3 & \boldsymbol{Q}_2 & \boldsymbol{Q}_1 \end{bmatrix} \tag{3.85}$$

The quaternion-valued vectors can be denoted as

$$\begin{aligned}
\bar{\boldsymbol{w}}^H &\triangleq [\boldsymbol{w}_1^T \quad -\boldsymbol{w}_2^T \quad -\boldsymbol{w}_3^T \quad -\boldsymbol{w}_4^T] \\
\boldsymbol{b}_{01}^1 &\triangleq [\boldsymbol{b}_{01,1}^T \quad -\boldsymbol{b}_{01,2}^T \quad -\boldsymbol{b}_{01,3}^T \quad -\boldsymbol{b}_{01,4}^T]^T \\
\boldsymbol{b}_{01}^2 &\triangleq [\boldsymbol{b}_{01,2}^T \quad \boldsymbol{b}_{01,1}^T \quad \boldsymbol{b}_{01,4}^T \quad -\boldsymbol{b}_{01,3}^T]^T \\
\boldsymbol{b}_{01}^3 &\triangleq [\boldsymbol{b}_{01,3}^T \quad -\boldsymbol{b}_{01,4}^T \quad \boldsymbol{b}_{01,1}^T \quad \boldsymbol{b}_{01,2}^T]^T \\
\boldsymbol{b}_{01}^4 &\triangleq [\boldsymbol{b}_{01,4}^T \quad \boldsymbol{b}_{01,3}^T \quad -\boldsymbol{b}_{01,2}^T \quad \boldsymbol{b}_{01,1}^T]^T \\
\boldsymbol{b}_{02}^1 &\triangleq [\boldsymbol{b}_{02,1}^T \quad -\boldsymbol{b}_{02,2}^T \quad -\boldsymbol{b}_{02,3}^T \quad -\boldsymbol{b}_{02,4}^T]^T \\
\boldsymbol{b}_{02}^2 &\triangleq [\boldsymbol{b}_{02,2}^T \quad \boldsymbol{b}_{02,1}^T \quad \boldsymbol{b}_{02,4}^T \quad -\boldsymbol{b}_{02,3}^T]^T \\
\boldsymbol{b}_{02}^3 &\triangleq [\boldsymbol{b}_{02,3}^T \quad -\boldsymbol{b}_{02,4}^T \quad \boldsymbol{b}_{02,1}^T \quad \boldsymbol{b}_{02,2}^T]^T \\
\boldsymbol{b}_{02}^4 &\triangleq [\boldsymbol{b}_{02,4}^T \quad \boldsymbol{b}_{02,3}^T \quad -\boldsymbol{b}_{02,2}^T \quad \boldsymbol{b}_{02,1}^T]^T
\end{aligned} \tag{3.86}$$

It can be verified that

$$\begin{aligned}
||\boldsymbol{w}^H \boldsymbol{Q}|| &= ||\bar{\boldsymbol{w}}^H \bar{\boldsymbol{Q}}|| \\
Re\{\boldsymbol{w}^H \boldsymbol{b}_{01}\} = \bar{\boldsymbol{w}}^H \boldsymbol{b}_{01}^1 \quad &, \quad Re\{\boldsymbol{w}^H \boldsymbol{b}_{02}\} = \bar{\boldsymbol{w}}^H \boldsymbol{b}_{02}^1 \\
Im^{(i)}\{\boldsymbol{w}^H \boldsymbol{b}_{01}\} = \bar{\boldsymbol{w}}^H \boldsymbol{b}_{01}^2 \quad &, \quad Im^{(i)}\{\boldsymbol{w}^H \boldsymbol{b}_{02}\} = \bar{\boldsymbol{w}}^H \boldsymbol{b}_{02}^2 \\
Im^{(j)}\{\boldsymbol{w}^H \boldsymbol{b}_{01}\} = \bar{\boldsymbol{w}}^H \boldsymbol{b}_{01}^3 \quad &, \quad Im^{(j)}\{\boldsymbol{w}^H \boldsymbol{b}_{02}\} = \bar{\boldsymbol{w}}^H \boldsymbol{b}_{02}^3 \\
Im^{(k)}\{\boldsymbol{w}^H \boldsymbol{b}_{01}\} = \bar{\boldsymbol{w}}^H \boldsymbol{b}_{01}^4 \quad &, \quad Im^{(k)}\{\boldsymbol{w}^H \boldsymbol{b}_{02}\} = \bar{\boldsymbol{w}}^H \boldsymbol{b}_{02}^4
\end{aligned} \tag{3.87}$$

Based on the above real-valued matrices and vectors, the formulation changes to

$$\begin{aligned} & \min ||\bar{\boldsymbol{w}}^H \bar{\boldsymbol{Q}}^H|| \\ \text{s.t. } & \bar{\boldsymbol{w}}^H \boldsymbol{b}_{01}^1 \geq 1 + \varepsilon ||\boldsymbol{w}|| \quad , \quad \bar{\boldsymbol{w}}^H \boldsymbol{b}_{02}^1 = 0 \\ & \bar{\boldsymbol{w}}^H \boldsymbol{b}_{01}^2 = 0 \quad , \quad \bar{\boldsymbol{w}}^H \boldsymbol{b}_{02}^2 \geq 1 + \varepsilon ||\boldsymbol{w}|| \\ & \bar{\boldsymbol{w}}^H \boldsymbol{b}_{01}^3 = 0 \quad , \quad \bar{\boldsymbol{w}}^H \boldsymbol{b}_{02}^3 = 0 \\ & \bar{\boldsymbol{w}}^H \boldsymbol{b}_{01}^4 = 0 \quad , \quad \bar{\boldsymbol{w}}^H \boldsymbol{b}_{02}^4 = 0 \end{aligned} \tag{3.88}$$

Solving the above convex optimization problem, we can obtain a real-valued weight vector $\bar{\boldsymbol{w}}^H \in \mathbb{R}^{1\times 4N}$. The quaternion-valued weight vector can be recovered from the four corresponding elements of $\bar{\boldsymbol{w}}^H$ by (3.84) and (3.86).

3.6 Simulations Results

In this section, simulations results are provided in terms of the resultant beam pattern, output SINR and robustness against steering vector mismatch. In the beam pattern part, both 2-D and 3-D beam patterns are presented for the proposed full Q-Capon beamformer to show the unity response to the desired signal and effective suppression to interferences. The beam pattern of the FQWCCB is also provided to show its tolerance around the desired DOA and polarisation region. In the output SINR performance part, the full Q-Capon beamformer, FQWCCB, and the Q-Capon beamformer are compared in two scenarios: one without steering vector mismatch, and one with steering vector mismatch. A solid-line is also displayed as the optimal (ideal) beamforming result. The error constant ε for FQWCCB is set to 1.3. In the robustness part, the beamformers are compared in terms of output SINR versus the snapshots number with 1° and 5° DOA and polarisation mismatch errors, respectively.

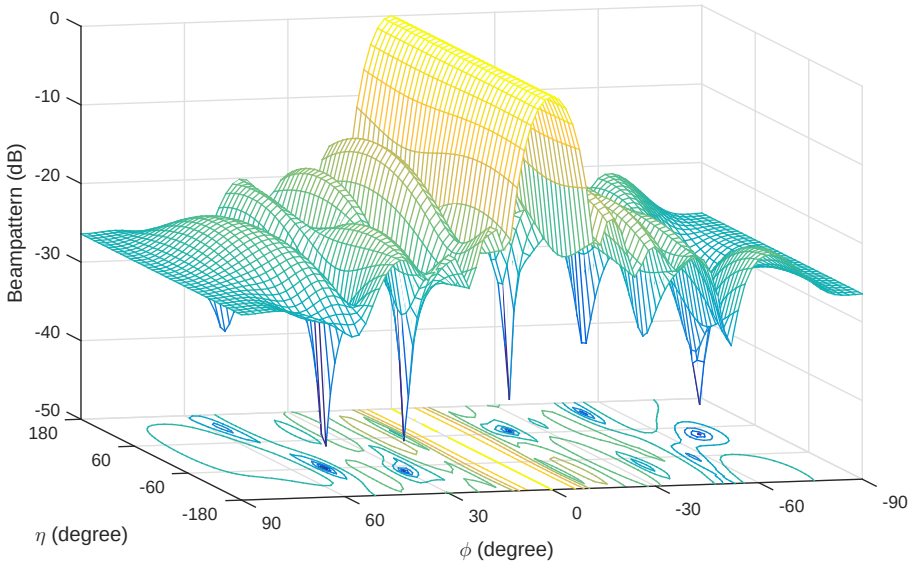

Figure 3.2: Beam pattern obtained from the full Q-Capon beamformer with $\theta = 90°$ and $\gamma = 60°$.

3.6.1 Beam pattern

In this part, result based on the algorithm is given to show interferences are suppressed in 3-D beampattern. In our simulations, we consider 10 pairs of cross-dipoles with half wavelength spacing. All signals are assumed to arrive from the same plane of $\theta = 90°$ and all interferences have the same polarisation parameter $\gamma = 60°$. For the SOI, the two sub-signals are set to $(90°, 1.5°, 90°, 45°)$ and $(90°, 1.5°, 0°, 0°)$, with interferences coming from $(90°, 30°, 60°, -80°)$, $(90°, -70°, 60°, 30°)$, $(90°, -20°, 60°, 70°)$, $(90°, 50°, 60°, -50°)$, respectively. The background noise is zero-mean quaternion-valued Gaussian. The power of SOI and all interfering signals are set equal and SNR (INR) is 20dB.

Fig. 3.2 shows the resultant 3-D beam pattern by the proposed full Q-Capon beamformer, where the interfering signals from (ϕ, η)=$(30°, -80°)$, $(-70°, 30°)$, $(-20°, 70°)$ and $(50°, -50°)$ have all been effectively suppressed, while the gain of SOI from $\phi = 1.5°$ stays almost a constant.

Figs. 3.3 and 3.4 show the 2-D beam patterns for $\eta = -80°, 30°, 70°, -50°$, respectively,

where d

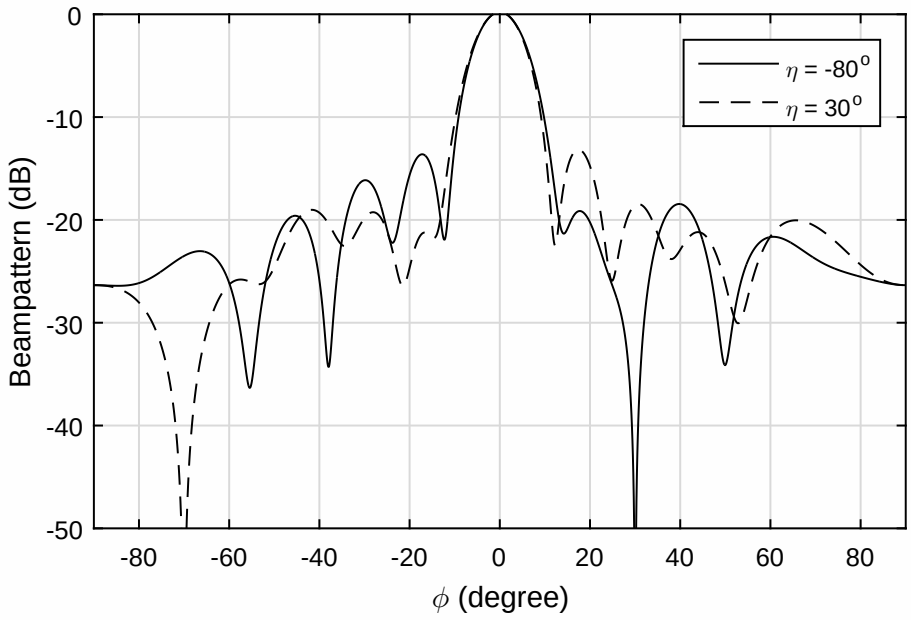

Figure 3.3: The resultant beam pattern for 1) $\theta = 90°, \gamma = 60°, \eta = -80°$; 2) $\theta = 90°, \gamma = 60°, \eta = 30°$.

The 3-D beam pattern for the proposed FQWCCB is provided in Fig. 3.5. We can still see that the interferences from (ϕ, η)=$(30°, -80°), (-70°, 30°), (-20°, 70°)$ and $(50°, -50°)$ are suppressed but the suppression is not as deep as in the case of the full Q-Capon beamformer; moreover, a large magnitude response are formed around the direction of the desired signal.

3.6.2 Output SINR performance

In the following, the output SINR performance of the two Capon beamformers (full Q-Capon and Q-Capon) is studied with the DOA and polarisation $(90°, 1.5°, 90°, 45°)$ and $(90°, 1.5°, 0°, 0°)$ for SOI and $(90°, 30°, 60°, -80°)$, $(90°, -70°, 60°, 30°)$, $(90°, -20°, 60°, 70°)$, $(90°, 50°, 60°, -50°)$ for interferences, respectively. Again, we have set SNR=INR=20dB. All results are obtained by averaging 1000 Monte-Carlo trials.

Firstly, we consider a scenario without steering vector mismatch. Fig. 3.6 shows the output SINR performance versus SNR with 100 snapshots, where the solid-line is for the

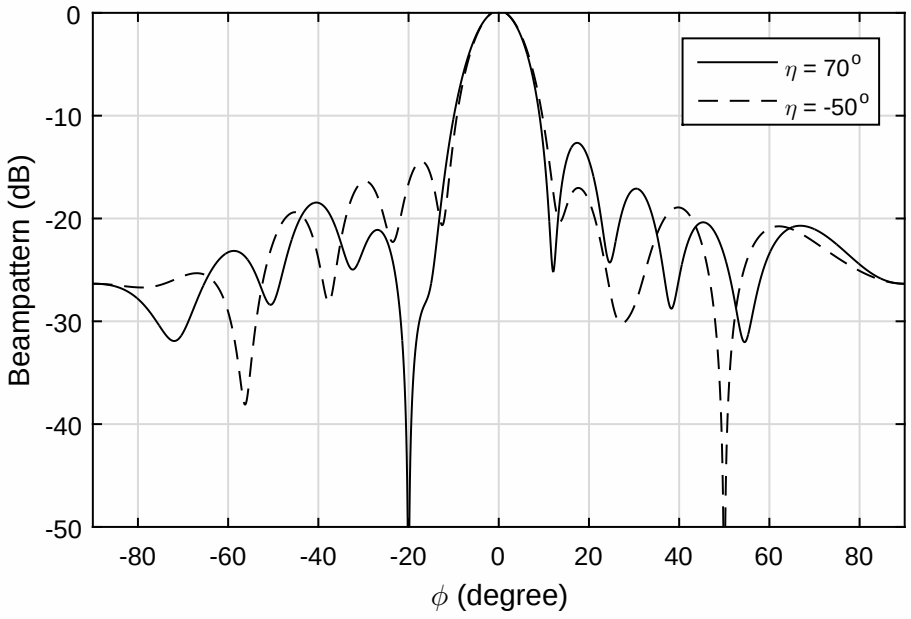

Figure 3.4: The resultant beam pattern for 1) $\theta = 90°, \gamma = 60°, \eta = 70°$; 2) $\theta = 90°, \gamma = 60°, \eta = -50°$.

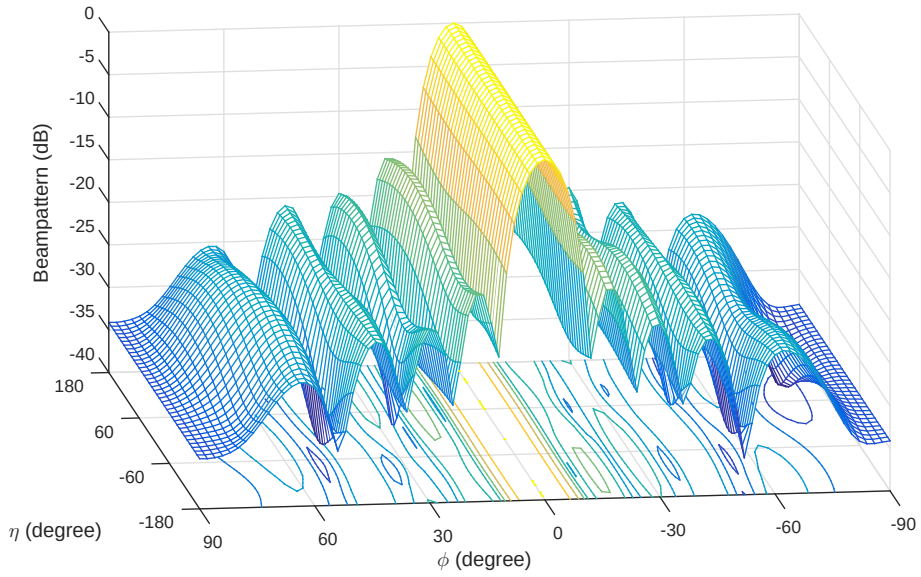

Figure 3.5: The resultant FQWCCB beam pattern with $\theta = 90°$ and $\gamma = 60°$.

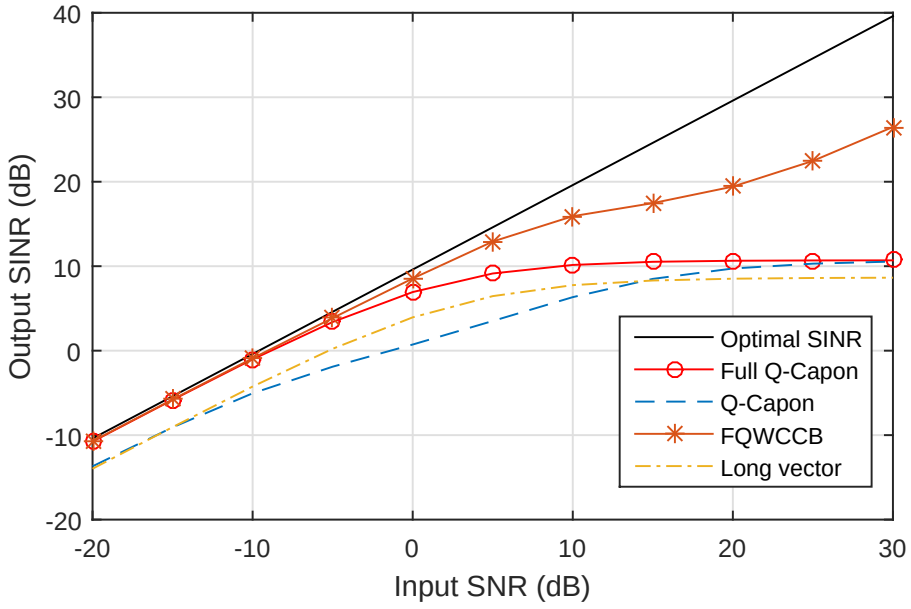

Figure 3.6: Output SINR versus input SNR, for snapshots number 100, without steering vector mismatch.

optimal beamformer, i.e. with infinite number of snapshots. Note that with only 100 snapshots, there will be non-negligible data model errors. It can be seen that for most of the input SNR range, the FQWCCB has the best output SINR performance versus the input SNR, while the full Q-Capon has a worse performance, but it still outperforms the Q-Capon beamformer. In the lower SNR range, FQWCCB and the full Q-Capon beamformer have a similar result, but the gap becomes greater from input SNR=10dB onwards. Also in the lower range, the proposed full Q-Capon beamformer has a better performance than the Q-Capon beamformer; for very high input SNR values, these two beamformers have a very similar performance. The long vector beamformer has a similar performance with Q-Capon beamformer. With the increase of SNR, when SNR reaches 0 to 10 dB, the long vector beamformer outperforms the Q-Capon beamformer. However, when SNR continues to increase to above 15 dB, the long vector beamformer loses its advantage against Q-Capon beamformer.

Next, another set of simulations is performed in a similar setting, which is shown in Fig. 3.7. The only difference is that there is 5° DOA and polarisation mismatch

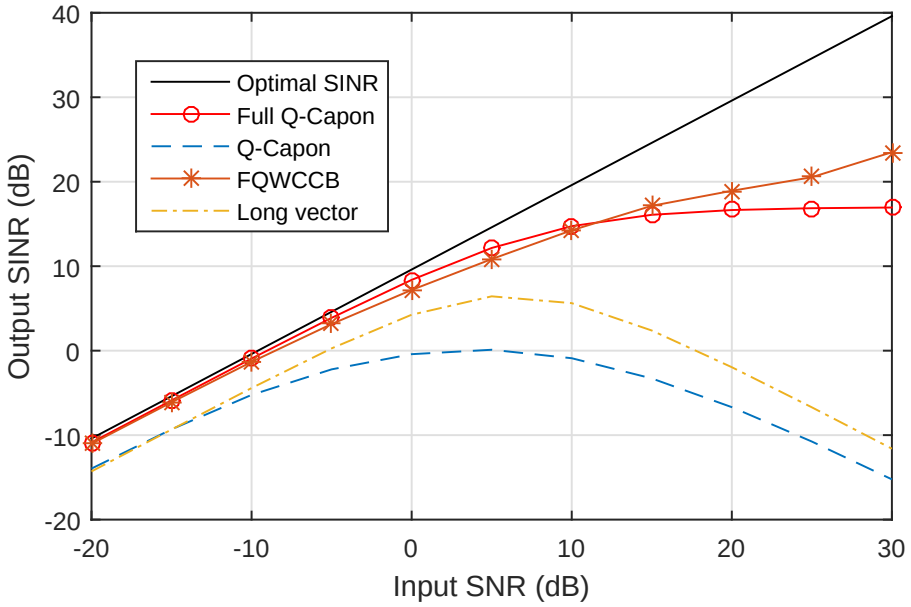

Figure 3.7: Output SINR versus input SNR, for snapshots number 100, with 5° mismatch.

for the SOI: the actual parameters of the two desired sub-signals are $(95°, 6.5°, 95°, 50°)$ and $(95°, 6.5°, 5°, 5°)$. For lower input SNR values (from -20 dB to -10 dB), the four beamformers have a similar output SINR, all worse than the example without steering vector mismatch. As the input SNR increases, difference among the three beamformers appears. From SNR=0dB onwards, the output SINR of FQWCCB and the full Q-Capon beamformer continue to increase while the output SINR of the Q-Capon beamformer and long vector beamformer start to drop slowly. Compared with the full Q-Capon beamformer, the FQWCCB has achieved a higher output SINR than the full Q-Capon beamformer with high SNR, which means the FQWCCB has a better performance than the full Q-Capon beamformer in this situation.

3.6.3 Performance with DOA and polarisation mismatch

Next, we investigate their performance in the presence of both DOA and polarisation errors with SNR=SIR=15dB. The output SINR with respect to the number of snapshots is shown in Fig. 3.8 in the presence of 1° error for the SOI, where the actual DOA and

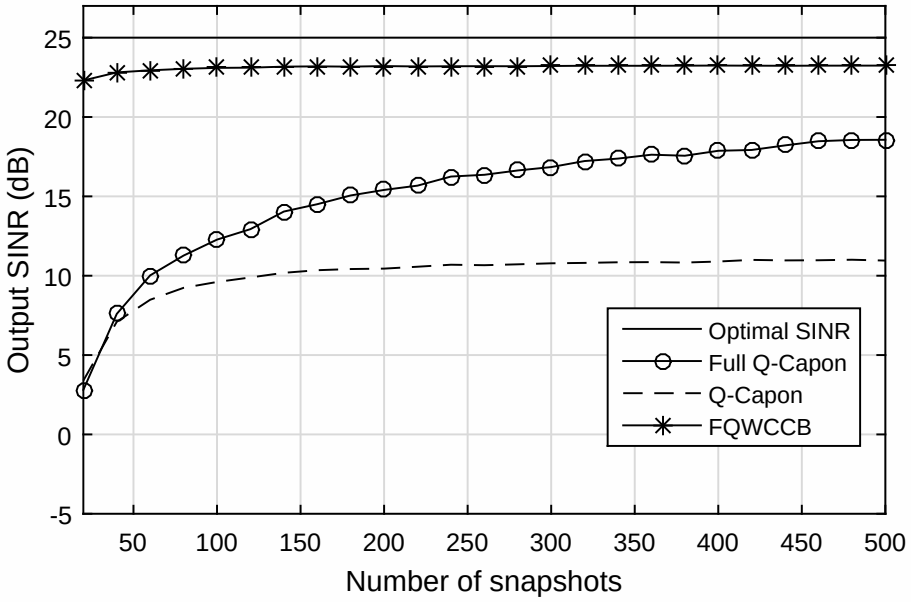

Figure 3.8: Output SINR versus snapshot number with SNR=SIR=15dB and 1° error.

polarisation parameters are $(91°, 2.5°, 91°, 46°)$ and $(91°, 2.5°, 1°, 1°)$.

We can see that the FQWCCB has the best performance in the presence of DOA and polarisation mismatch, with output SINR above 20dB for a varied number of snapshots. Both the full Q-Capon beamformer and the Q-Capon beamformer experience a loss of SINR (below 5dB at the beginning and increases with the snapshots number). However, the full Q-Capon beamformer still has achieved a much higher output SINR than the Q-Capon beamformer, and this gap increases with the increase of snapshots number. Fig. 3.9 shows a similar trend in the presence of a 5° error. For the Q-Capon beamformer, the output SINR stays below 5dB with little fluctuation, while the full Q-Capon beamformer and its worst-case constraint version can achieve a relatively high output SINR level of above 15dB with 500 snapshots. Overall, we can see that the proposed full Q-Capon beamformer (both with worst-case constraint and without worst-case constraint) is more robust against array steering vector errors than the original Q-Capon beamformer.

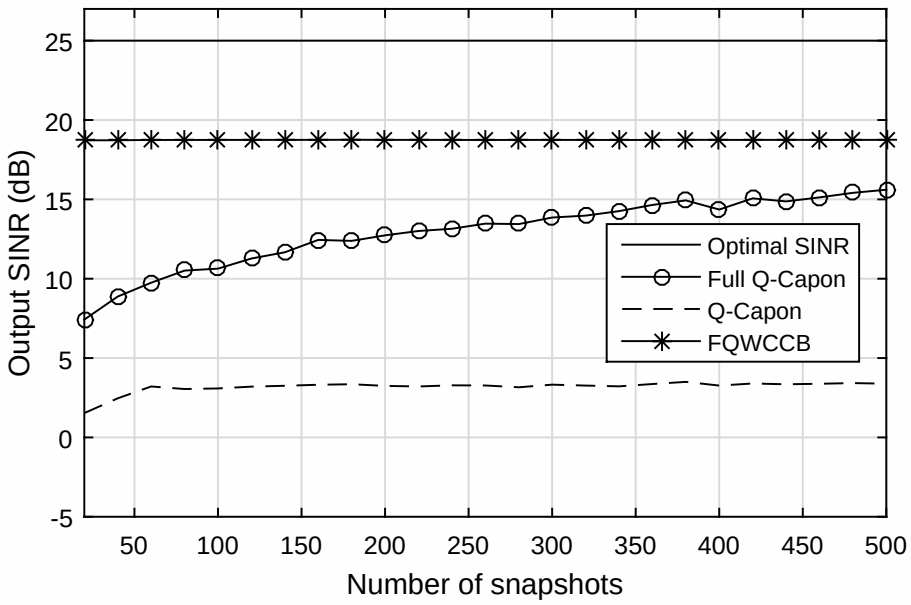

Figure 3.9: Output SINR versus snapshot number with SNR=SIR=15dB and 5° error.

3.7 Summary

In this chapter, a full quaternion model has been developed for adaptive beamforming based on crossed-dipole vector sensor arrays, with a new full quaternion Capon beamformer derived. Different from previous studies in quaternion-valued adaptive beamforming, we have considered a quaternion-valued desired signal, given the recent research development in quaternion-valued communications. The proposed beamformer has a better performance and a much lower computational complexity than a previously proposed Q-Capon beamformer and is also shown to be more robust against array pointing errors. Furthermore, based on this full quaternion model, the robust beamforming problem has also been studied in the presence of steering vector errors and a worst-case based robust beamformer was developed. The effectiveness of the full Q-Capon beamformer and the robustness of the further developed worst-case based beamformer was verified by simulations.

Chapter 4

Joint 4-D DOA and Polarization Estimation with Crossed-dipole and Tripole Sensor Arrays

4.1 Introduction

The joint estimation of direction of arrival and polarisation for signals based on electromagnetic (EM) vector sensor arrays has been widely studied in the past [34, 61–66]. In [34], the EM vector sensor was first used to collect both electric and magnetic information of the impinging signals, where all six electromagnetic components are measured to identify the signals. So far most of the studies are focused on the linear array structure employing crossed-dipoles, where the general two-dimensional (2-D) DOA model is simplified into a one-dimensional (1-D) one by assuming that all the signals arrive from the same known azimuth angle ϕ. In [43], a quaternion MUSIC algorithm was proposed to deal with the joint DOA (θ) and polarisation (ρ, ϕ) estimation problem by considering the two complex-valued signals received by each crossed-dipole sensor as the four elements of a quaternion, where a three-dimensional (3-D) peak search is required with a very high computational complexity. In [44], a quaternion ESPRIT algorithm was developed for

direction finding with a reduced complexity. Furthermore, a dimension-reduction MUSIC algorithm based on uniform linear arrays with crossed-dipole sensors was introduced in [67], where the 3-D joint peak search is replaced by a 1-D DOA search and a 2-D polarisation search.

In practice, the azimuth angle θ and the elevation angle ϕ of the signals are unknown and they are usually different for different signals and need to be estimated together. However, as rigorously proved for the first time in this chapter, linear crossed-dipole array cannot achieve this because such an array will produce an ambiguity problem, where the azimuth angle and the elevation angle of the impinging signals can not be uniquely identified and there could also be false peaks in the resultant spatial spectrum. To tackle this ambiguity problem, one solution is to extend the linear geometry to a two-dimensional (2-D) rectangular planar array, such as the uniform rectangular array (URA), at significant space cost, and one good example for this solution is the work presented in [68], where based on a URA, a pencil-MUSIC algorithm is proposed to solve the full 4-D DOA and polarisation estimation problem. However, it is not always feasible to use the rectangular array as a solution due to space limit. On the other hand, it is possible to add one dipole to the crossed-dipole structure to form a tripole sensor and tripole sensor array has been proposed in the past for DOA estimation [69–71]. Therefore, as another solution, motivated by simultaneously simplifying the array structure and reducing the computational complexity, instead of extending the linear crossed-dipole array to a higher spatial dimension, we replaced the crossed-dipoles by tripoles and constructed a linear tripole array in our earlier conference publication for joint 4-D DOA and polarisation estimation for the first time [72] . Moreover, for the first time, we give a clear proof about why a linear tripole array can avoid the ambiguity problem except for some special cases and can be used for four-dimensional joint DOA and polarisation estimation.

At the algorithm level, two MUSIC-like algorithms for the 4-D estimation problem are proposed. The first is a direct search in the 4-D space to locate the DOA and polarisation parameters simultaneously (4-D MUSIC), which has an extremely high computational

complexity. The other algorithm is to transform the 4-D search into two separate 2-D searches (2-D MUSIC), significantly reducing the computational complexity. To evaluate the performance of the proposed algorithms, the Cramér-Rao Bound (CRB) of the linear tripole array for 4-D estimation is derived. In the past, CRBs have been derived under different circumstances, such as the results for arrays with arbitrary geometries in [73]. Obviously, the types of signals and noise will affect the derived CRB result. Normally, noise is assumed to be temporally and spatially white and the source signal can have two different types: one is to assume the source signal is deterministic [74–76], while the other assumes that the signal is random and a common choice is being Gaussion distributed [77–82]. In this thesis, we assume the source signal is of the second type.

This chapter is structured as follows. The linear tripole array is introduced in Section II with a detailed proof for the 4-D ambiguity problem associated with the linear crossed-dipole array and why the linear tripole array can solve the problem. The two 4-D estimation algorithms are proposed in Section III with the CRB derived in detail. Simulation results are presented in Section IV, followed by a summary in Section V.

4.2 Tripole Sensor Array Model

4.2.1 Tripole sensor array

Suppose there are M uncorrelated narrowband signals impinging upon a uniform linear array with N tripoles, where each tripole consists of three co-located mutually perpendicular dipoles, as shown in Fig. 4.1. Assume that all signals are ellipse-polarized. The parameters, including DOA and polarisation of the m-th signal are denoted by $(\theta_m, \phi_m, \gamma_m, \eta_m), m = 1, 2, ..., M$. The inter-element spacing d of the array is $\lambda/2$, where λ is the wavelength of the incoming signals. For each tripole sensor, the three components are parallel to x, y and z axes, respectively. The background noise is Gaussian white noise with zero mean and variance σ_n^2, which is uncorrelated with the impinging signals. Due to the phase shift among the sensors, the steering vector for the m-th signal can be

denoted as

$$\boldsymbol{a}_m = [1, e^{-j\pi \sin\theta_m \sin\phi_m}, ..., e^{-j(N-1)\pi \sin\theta_m \sin\phi_m}] \tag{4.1}$$

and the polarisation vector $\boldsymbol{p}_m$ is determined by the product of DOA component $\boldsymbol{\Omega}_m$ and the polarization component $\boldsymbol{g}_m$, where

$$\boldsymbol{p}_m = \boldsymbol{\Omega}_m \boldsymbol{g}_m \tag{4.2}$$

The DOA component is a matrix consisting of two vectors that are orthogonal to the signal direction. There are infinite number of choices for these two vectors and generally the following two are used [83]

$$\boldsymbol{\Omega}_m = \begin{bmatrix} \cos\theta_m \cos\phi_m & -\sin\phi_m \\ \cos\theta_m \sin\phi_m & \cos\phi_m \\ -\sin\theta_m & 0 \end{bmatrix} \tag{4.3}$$

The corresponding polarization component is given by

$$\boldsymbol{g}_m = \begin{bmatrix} \sin\gamma_m e^{j\eta_m} \\ \cos\gamma_m \end{bmatrix} \tag{4.4}$$

where γ_m is the auxiliary polarization angle and η_m the polarization phase difference. By expanding (4.2), the polarisation vector $\boldsymbol{p}_m$ can be divided into three different components in x, y and z axes

$$\boldsymbol{p}_m = \begin{bmatrix} \cos\theta_m \cos\phi_m \sin\gamma_m e^{j\eta_m} - \sin\phi_m \cos\gamma_m \\ \cos\theta_m \sin\phi_m \sin\gamma_m e^{j\eta_m} + \cos\phi_m \cos\gamma_m \\ -\sin\theta_m \sin\gamma_m e^{j\eta_m} \end{bmatrix} \tag{4.5}$$

For convenience, we replace the three elements in $\boldsymbol{p}_m$ by p_{mx}, p_{my} and p_{mz}, which is given by:

$$\begin{aligned} p_{mx} &= \cos\theta_m \cos\phi_m \sin\gamma_m e^{j\eta_m} - \sin\phi_m \cos\gamma_m \\ p_{my} &= \cos\theta_m \sin\phi_m \sin\gamma_m e^{j\eta_m} + \cos\phi_m \cos\gamma_m \\ p_{mz} &= -\sin\theta_m \sin\gamma_m e^{j\eta_m} \end{aligned} \tag{4.6}$$

The received signal at the tripole sensor array can be denoted as a function of steering vector $\boldsymbol{a}_m$, polarisation vector $\boldsymbol{p}_m$, source signals s_m and background noise $\boldsymbol{n}$. At the k-th time instant, the received signal vector $\boldsymbol{x}[k]$ can be expressed as

$$\begin{aligned} \boldsymbol{x}[k] &= \sum_{m=1}^{M} [\boldsymbol{a}_m \otimes \boldsymbol{p}_m] s_m[k] + \boldsymbol{n}[k] \\ &= \sum_{m=1}^{M} \boldsymbol{v}_m s_m[k] + \boldsymbol{n}[k] \end{aligned} \tag{4.7}$$

where '$\otimes$' stands for the Kronecker product, $\boldsymbol{v}_m$ is the Kronecker product of $\boldsymbol{a}_m$ and $\boldsymbol{p}_m$, and $\boldsymbol{n}[k]$ is a $3N \times 1$ Gaussian white noise vector. The covariance matrix $\boldsymbol{R}$ of the received signal vector is given by

$$\begin{aligned} \boldsymbol{R} &= E\{\boldsymbol{x}[k]\boldsymbol{x}[k]^H\} \\ &= \sum_{m=1}^{M} \boldsymbol{v}_m s[k] s[k]^H \boldsymbol{v}_m^H + \sigma_n^2 \boldsymbol{I}_{3N} \end{aligned} \tag{4.8}$$

In practice, $\boldsymbol{R}$ is not available and can be estimated by averaging a finite number of snapshots. In such case, an estimated covariance matrix $\hat{\boldsymbol{R}}$ is used to replace $\boldsymbol{R}$

$$\hat{\boldsymbol{R}} \approx \frac{1}{K} \sum_{l=1}^{L} \boldsymbol{x}[k]\boldsymbol{x}[k]^H \tag{4.9}$$

where K is the number of snapshots.

4.2.2 Comparison between crossed-dipole array and tripole array

This section will mainly show why the ULA with crossed-dipoles cannot uniquely determine the four parameters associated with each impinging signal, leading to the spatial aliasing problem, and why the ULA with tripoles can provide a unique solution for the joint 4-D estimation problem.

To show the ambiguity problem, we consider one source signal impinging upon the array so that the subscript m can be dropped for convenience. The joint DOA and

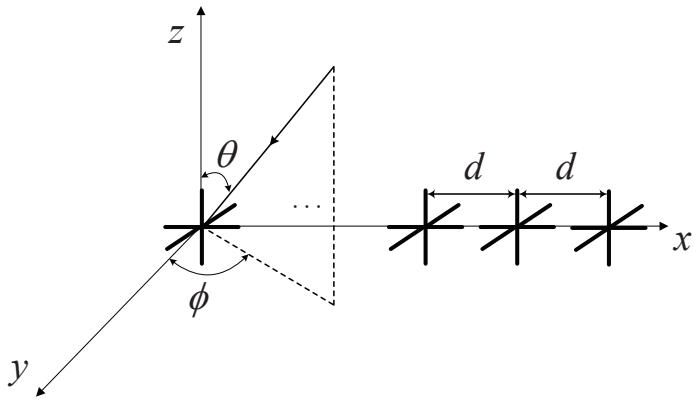

Figure 4.1: Geometry of a uniform linear tripole array, where a signal arrives from elevation angle θ and azimuth angle ϕ.

polarisation estimation problem can be considered as estimation of the steering vector of this source signal.

For crossed-dipole sensor array, its joint steering vector $\boldsymbol{w}$ is given by

$$\boldsymbol{w} = \boldsymbol{a} \otimes \boldsymbol{q} \tag{4.10}$$

where

$$\boldsymbol{q} = \begin{bmatrix} p_x \\ p_y \end{bmatrix} \tag{4.11}$$

Here, $\boldsymbol{w}$ is a $2N \times 1$ vector with a 2×1 polarisation vector $\boldsymbol{q}$. For the tripole sensor array, the joint steering vector $\boldsymbol{v}$ is a $3N \times 1$ vector with a 3×1 polarisation vector $\boldsymbol{p}$, i.e.

$$\boldsymbol{v} = \boldsymbol{a} \otimes \boldsymbol{p} \tag{4.12}$$

where

$$\boldsymbol{p} = \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} \tag{4.13}$$

The ambiguity problem associated with the cross-dipole array can be stated as follows: for a given steering vector $\boldsymbol{w}_1$, by varying parameters $(\theta, \phi, \gamma, \eta)$, there exists at least one

vector $\boldsymbol{w}_2$ which is in parallel with $\boldsymbol{w}_1$. By parallel we mean

$$\boldsymbol{w}_2 = k \cdot \boldsymbol{w}_1 \tag{4.14}$$

where k is an arbitrary complex-valued scalar.

When we say that the tripole array can avoid the ambiguity problem, we mean that for a given tripole array steering vector $\boldsymbol{v}_1$, no matter how we change the parameters $(\theta, \phi, \gamma, \eta)$ change, it is impossible to find another steering vector $\boldsymbol{v}_2$ which is in parallel with $\boldsymbol{v}_1$.

To prove these two statements, firstly we give the following lemma.

Lemma 1 *Given two complex-valued vectors* $\mathbf{w}_1 = \mathbf{a}_1 \otimes \mathbf{q}_1$ *and* $\mathbf{w}_2 = \mathbf{a}_2 \otimes \mathbf{q}_2$, $\mathbf{w}_1 // \mathbf{w}_2$ *is necessary and sufficient for* $\mathbf{a}_1 // \mathbf{a}_2$ *and* $\mathbf{q}_1 // \mathbf{q}_2$, *where // means the two vectors are in parallel.*

Proof 4.2.1 *Necessity: If* $\mathbf{a}_1 // \mathbf{a}_2$ *and* $\mathbf{q}_1 // \mathbf{q}_2$, *then*

$$\begin{aligned} \mathbf{a}_2 &= k_1 \cdot \mathbf{a}_1 \\ \mathbf{q}_2 &= k_2 \cdot \mathbf{q}_1 \end{aligned} \tag{4.15}$$

where k_1 *and* k_2 *are arbitrary complex-valued constants. Then,*

$$\begin{aligned} \mathbf{w}_2 &= \mathbf{a}_2 \otimes \mathbf{q}_2 \\ &= (k_1 \cdot \mathbf{a}_1) \otimes (k_2 \cdot \mathbf{q}_1) \\ &= (k_1 k_2) \cdot (\mathbf{a}_1 \otimes \mathbf{q}_1) \\ &= (k_1 k_2) \cdot \mathbf{w}_1 \end{aligned} \tag{4.16}$$

Hence, $\mathbf{w}_1 // \mathbf{w}_2$.

Sufficiency: By (4.10), $\mathbf{w}$ *can be expanded as*

$$\mathbf{w} = \mathbf{a} \otimes \mathbf{q} = \begin{bmatrix} a_1\mathbf{q} \\ . \\ . \\ . \\ a_N\mathbf{q} \end{bmatrix} = \begin{bmatrix} a_1 p_x \\ a_1 p_y \\ . \\ . \\ . \\ a_N p_x \\ a_N p_y \end{bmatrix} \tag{4.17}$$

The Hermitian transpose $\mathbf{w}^H$ *is given by*

$$\mathbf{w}^H = \mathbf{a}^H \otimes \mathbf{q}^H \tag{4.18}$$

The norm of $\mathbf{w}$ *is*

$$\begin{aligned} |\mathbf{w}| &= \sqrt{\mathbf{w}^H \mathbf{w}} \\ &= \sqrt{(a_1 a_1^* + ... + a_N a_N^*)(p_x p_x^* + p_y p_y^*)} \\ &= |\mathbf{a}| \cdot |\mathbf{q}| \end{aligned} \tag{4.19}$$

Then, we have

$$\begin{aligned} |\mathbf{w}_1| &= |\mathbf{a}_1| \cdot |\mathbf{q}_1| \\ |\mathbf{w}_2| &= |\mathbf{a}_2| \cdot |\mathbf{q}_2| \end{aligned} \tag{4.20}$$

Generally, by (4.18), the modulus of the inner product of $\mathbf{w}_1$ *and* $\mathbf{w}_2$ *can be expanded as*

$$|\mathbf{w}_1^H \mathbf{w}_2| = |(\mathbf{a}_1^H \otimes \mathbf{q}_1^H) \cdot (\mathbf{a}_2 \otimes \mathbf{q}_2)| \tag{4.21}$$

According to the mixed-product property of Kronecker product, the lemma 4.2.10 in [84], (4.21) can be deduced to

$$\begin{aligned} |\mathbf{w}_1^H \mathbf{w}_2| &= |\mathbf{a}_1^H \cdot \mathbf{a}_2| \otimes |\mathbf{q}_1^H \cdot \mathbf{q}_2| \\ &\leq |\mathbf{a}_1| \cdot |\mathbf{a}_2| \cdot |\mathbf{q}_1| \cdot |\mathbf{q}_2| \end{aligned} \tag{4.22}$$

On the other hand, since $\mathbf{w}_1//\mathbf{w}_2$*, we know* $\mathbf{w}_2 = k\mathbf{w}_1$ *and* $|\mathbf{w}_2| = |k||\mathbf{w}_1|$*, which leads to*

$$\begin{aligned} |\mathbf{w}_1^H \mathbf{w}_2| &= |\mathbf{w}_1^H \cdot k\mathbf{w}_1| = |k||\mathbf{w}_1| \cdot |\mathbf{w}_1| \\ &= |\mathbf{w}_1| \cdot |\mathbf{w}_2| = |\mathbf{a}_1| \cdot |\mathbf{a}_2| \cdot |\mathbf{q}_1| \cdot |\mathbf{q}_2| \end{aligned} \tag{4.23}$$

The equality in (4.22) holds only when $\mathbf{a}_1//\mathbf{a}_2$ *and* $\mathbf{q}_1//\mathbf{q}_2$*. Combined with (4.23), the sufficiency proof is completed.*

Although we used the joint steering vector of the crossed-dipole array in the above proof, it is straightforward to show that Lemma 1 is also applicable to the joint steering vector of tripole sensor arrays.

Now we first consider the ambiguity problem in crossed-dipole sensor arrays. Given $\boldsymbol{w}_1 = \boldsymbol{a}_1 \otimes \boldsymbol{q}_1$, our aim is to find a vector $\boldsymbol{w}_2 = \boldsymbol{a}_2 \otimes \boldsymbol{q}_2$ with $\boldsymbol{a}_1//\boldsymbol{a}_2$ and $\boldsymbol{q}_1//\boldsymbol{q}_2$.

First Equation (4.1) indicates that $\boldsymbol{a}_1$ is only determined by the value of $\sin\theta_1 \sin\phi_1$. To ensure $\boldsymbol{a}_1//\boldsymbol{a}_2$, given an arbitrary value from 0^o to 90^o for θ_2, ϕ_2 can be worked out by

$$\sin\theta_1 \sin\phi_1 = \sin\theta_2 \sin\phi_2 \tag{4.24}$$

Based on (4.24), we need further choose values for γ_2 and η_2 to satisfy $\boldsymbol{q}_1//\boldsymbol{q}_2$. From (4.6) and (4.11), the polarisation vector $\boldsymbol{q}_1$ is determined by all four parameters θ_1, ϕ_1, γ_1 and η_1 with

$$\begin{aligned} \boldsymbol{q}_1 &= \begin{bmatrix} \cos\theta_1 \cos\phi_1 & -\sin\phi_1 \\ \cos\theta_1 \sin\phi_1 & \cos\phi_1 \end{bmatrix} \begin{bmatrix} \sin\gamma_1 e^{j\eta_1} \\ \cos\gamma_1 \end{bmatrix} \\ &= \boldsymbol{\Psi}_1 \boldsymbol{g}_1 \end{aligned} \tag{4.25}$$

Hence, the other polarisation vector $\boldsymbol{q}_2 = \boldsymbol{\Psi}_2 \boldsymbol{g}_2$ need to satisfy

$$\begin{aligned} \boldsymbol{\Psi}_1 \boldsymbol{g}_1 &= \lambda \boldsymbol{\Psi}_2 \boldsymbol{g}_2 \\ \Rightarrow \boldsymbol{g}_2 &= \lambda^{-1} \boldsymbol{\Psi}_2^{-1} \boldsymbol{\Psi}_1 \boldsymbol{g}_1 \end{aligned} \tag{4.26}$$

λ is a constant and without loss of generality we assume its value is 1. Here $\boldsymbol{g}_2$ is a 2×1 vector with $\boldsymbol{g}_2[1] = \sin\gamma_2 e^{j\eta_2}$ and $\boldsymbol{g}_2[2] = \cos\gamma_2$, where "[1]" and "[2]" denote the first

and the second elements of the vector.

$$\tan\gamma_2 = \frac{|\boldsymbol{g}_2[1]|}{|\boldsymbol{g}_2[2]|}$$
$$\tan\eta_2 = \frac{\mathrm{Im}\{\boldsymbol{g}_2[1]/\boldsymbol{g}_2[2]\}}{\mathrm{Re}\{\boldsymbol{g}_2[1]/\boldsymbol{g}_2[2]\}} \tag{4.27}$$

Using the values of (4.24) and (4.27), the new joint steering vector $\boldsymbol{q}_2$ will be in parallel with the original $\boldsymbol{q}_1$. As a result, we can not uniquely determine the four DOA and polarisation parameters of a source using the crossed-dipole array.

Next, we consider the tripole sensor array case. Given a joint steering vector $\boldsymbol{v}_1 = \boldsymbol{a}_1 \otimes \boldsymbol{p}_1$, we want to prove that a parallel $\boldsymbol{v}_2 = \boldsymbol{a}_2 \otimes \boldsymbol{p}_2$ does not exist and we prove it by contradiction. Similar to the crossed-dipole case, an arbitrary value from 0^o to 90^o is chosen for θ_2, and then (4.24) is applied to calculate ϕ_2, so that the new elevation and azimuth angles ensure $\boldsymbol{a}_1 // \boldsymbol{a}_2$. For the polarisation vector $\boldsymbol{p}_1$, assume that there exists a vector $\boldsymbol{p}_2$ which is in parallel with $\boldsymbol{p}_1$, i.e.

$$\boldsymbol{\Omega}_1 \boldsymbol{g}_1 = \lambda \boldsymbol{\Omega}_2 \boldsymbol{g}_2 \tag{4.28}$$

where λ is an unknown complex-valued constant. Expanding $\boldsymbol{\Omega}_1$ and $\boldsymbol{\Omega}_2$ by the column vector, where $\boldsymbol{\Omega}_{11}$ and $\boldsymbol{\Omega}_{12}$ are the first and second column vectors of $\boldsymbol{\Omega}_1$, and $\boldsymbol{\Omega}_{21}$ and $\boldsymbol{\Omega}_{22}$ are the first and second column vectors of $\boldsymbol{\Omega}_2$, respectively. (4.28) is transformed to

$$[\boldsymbol{\Omega}_{11} \quad \boldsymbol{\Omega}_{12}] \begin{bmatrix} \boldsymbol{g}_1[1] \\ \boldsymbol{g}_1[2] \end{bmatrix} = \lambda [\boldsymbol{\Omega}_{21} \quad \boldsymbol{\Omega}_{22}] \begin{bmatrix} \boldsymbol{g}_2[1] \\ \boldsymbol{g}_2[2] \end{bmatrix}$$
$$\Updownarrow$$
$$\boldsymbol{\Omega}_{11}\boldsymbol{g}_1[1] + \boldsymbol{\Omega}_{12}\boldsymbol{g}_1[2] = \boldsymbol{\Omega}_{21}\boldsymbol{g}_2[1]\lambda + \boldsymbol{\Omega}_{22}\boldsymbol{g}_2[2]\lambda \tag{4.29}$$

The left side of (4.29) can be viewed as a vector which is a linear combination of $\boldsymbol{\Omega}_{11}$ and $\boldsymbol{\Omega}_{12}$. The right is a linear combination of $\boldsymbol{\Omega}_{21}$ and $\boldsymbol{\Omega}_{22}$. Here we define a two-dimensional space $\boldsymbol{A}_1$ spanned by $\boldsymbol{\Omega}_{11}$ and $\boldsymbol{\Omega}_{12}$, also $\boldsymbol{A}_2$ spanned by $\boldsymbol{\Omega}_{21}$ and $\boldsymbol{\Omega}_{22}$. Since $\boldsymbol{\Omega}_{11}, \boldsymbol{\Omega}_{12}, \boldsymbol{\Omega}_{21}$ and $\boldsymbol{\Omega}_{22}$ are all 3×1 vectors, the equation holds only in the following two cases:

Case 1: $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$ are the same two-dimensional span.

It can be noticed that $\boldsymbol{A}_1$ intersects $x-y$ plane at vector $\boldsymbol{\Omega}_{12}$. And $\boldsymbol{A}_2$ intersects $x-y$ plane at vector $\boldsymbol{\Omega}_{22}$. If $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$ are the same two-dimensional span, it must satisfy that $\boldsymbol{\Omega}_{12}//\boldsymbol{\Omega}_{22}$, then we have

$$\begin{bmatrix} -\sin\phi_1 \\ \cos\phi_1 \\ 0 \end{bmatrix} // \begin{bmatrix} -\sin\phi_2 \\ \cos\phi_2 \\ 0 \end{bmatrix} \Leftrightarrow -\frac{\sin\phi_1}{\cos\phi_1} = -\frac{\sin\phi_2}{\cos\phi_2}$$

$$\Leftrightarrow \tan\phi_1 = \tan\phi_2 \tag{4.30}$$

However, $\phi_1 \neq \phi_2$, (4.30) conflicts with the basic assumption, which means with tripole sensor array, there is no other joint steering vector $\boldsymbol{v}_2$ in parallel with the given $\boldsymbol{v}_1$ in such a case.

Case 2: $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$ are two different two-dimensional spans. Then $\boldsymbol{p}_1$ and $\boldsymbol{p}_2$ must be in parallel with the intersecting vector of $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$.

Firstly we denote the intersecting vector as $\boldsymbol{\Omega}_x$. Since $\boldsymbol{\Omega}_{11}, \boldsymbol{\Omega}_{12}, \boldsymbol{\Omega}_{21}, \boldsymbol{\Omega}_{22}$ are all real-valued vectors, all the elements in the intersection vector $\boldsymbol{\Omega}_x$ must be also real-valued. From eq.(4.5), $\boldsymbol{p}_1$ can be transformed to

$$\begin{aligned} \boldsymbol{p}_1 &= e^{j\eta} \begin{bmatrix} \cos\theta\cos\phi\sin\gamma - \sin\phi\cos\gamma e^{-j\eta} \\ \cos\theta\sin\phi\sin\gamma + \cos\phi\cos\gamma e^{-j\eta} \\ -\sin\theta\sin\gamma \end{bmatrix} \\ &= e^{j\eta} \cdot \hat{\boldsymbol{p}}_1 \end{aligned} \tag{4.31}$$

It can be seen that $\boldsymbol{p}_1//\hat{\boldsymbol{p}}_1$. In most situations, with $\gamma \neq 90°$, $\gamma \neq 0$ and $\eta \neq 0$, the first two elements in $\hat{\boldsymbol{p}}_1$ are complex-valued and the last element in $\hat{\boldsymbol{p}}_1$ is real-valued, which indicates that with such a situation, it is impossible for $\hat{\boldsymbol{p}}_1$ to be in parallel with the intersecting vector $\boldsymbol{\Omega}_x$. Hence, with $\gamma \neq 90°$, $\gamma \neq 0$ and $\eta \neq 0$, there is no ambiguity in DOA and polarisation estimation with tripole sensors.

However, when $\gamma = 90°$ or $\gamma = 0$ or $\eta = 0$, $\hat{\boldsymbol{p}}_1$ or $\boldsymbol{p}_1$ becomes a vector with all elements being real-valued, and it may be possible for $\boldsymbol{p}_1$ to be in parallel with the intersecting vector $\boldsymbol{\Omega}_x$. Now with the assumption $\boldsymbol{p}_1//\boldsymbol{p}_2//\boldsymbol{\Omega}_x$, $\boldsymbol{p}_1$ and $\boldsymbol{p}_2$ must all real-valued, which

means $\gamma_1 = 90^\circ$ or $\gamma_1 = 0$ or $\eta_1 = 0$, and at the same time $\gamma_2 = 90^\circ$ or $\gamma_2 = 0$ or $\eta_2 = 0$. With the constraint $\sin\theta_1 \sin\phi_1 = \sin\theta_2 \sin\phi_2$, we consider all of the nine different cases:

Case 2.1: $\gamma_1 = 90^\circ$ and $\gamma_2 = 90^\circ$.

In this case

$$\begin{aligned} \boldsymbol{p}_1 &= e^{j\eta_1} \begin{bmatrix} \cos\theta_1 \cos\phi_1 \\ \cos\theta_1 \sin\phi_1 \\ -\sin\theta_1 \end{bmatrix} \\ \boldsymbol{p}_2 &= e^{j\eta_2} \begin{bmatrix} \cos\theta_2 \cos\phi_2 \\ \cos\theta_2 \sin\phi_2 \\ -\sin\theta_2 \end{bmatrix} \end{aligned} \tag{4.32}$$

With $\theta_1 = \theta_2$ and $\phi_1 = \phi_2$, we have $\boldsymbol{p}_1 // \boldsymbol{p}_2$ for arbitrary η_1 and η_2. An example is $(30^\circ, 60^\circ, 90^\circ, 20^\circ)$ and $(30^\circ, 60^\circ, 90^\circ, 50^\circ)$.

Case 2.2: $\gamma_1 = 90^\circ$ and $\gamma_2 = 0^\circ$. (same for $\gamma_1 = 0^\circ$ and $\gamma_2 = 90^\circ$)

$$\begin{aligned} \boldsymbol{p}_1 &= e^{j\eta_1} \begin{bmatrix} \cos\theta_1 \cos\phi_1 \\ \cos\theta_1 \sin\phi_1 \\ -\sin\theta_1 \end{bmatrix} \\ \boldsymbol{p}_2 &= \begin{bmatrix} -\sin\phi_2 \\ \cos\phi_2 \\ 0 \end{bmatrix} \end{aligned} \tag{4.33}$$

In this case, with $\theta_1 = 0^\circ$ and $\tan\phi_1 = -\cot\phi_2$, we have $\boldsymbol{p}_1 // \boldsymbol{p}_2$ for arbitrary θ_2, η_1 and η_2. An example is $(0^\circ, 90^\circ, 90^\circ, 20^\circ)$ and $(50^\circ, 0^\circ, 0^\circ, 50^\circ)$.

Case 2.3: $\gamma_1 = 90^\circ$ and $\eta_2 = 0^\circ$. (same for $\eta_1 = 0^\circ$ and $\gamma_2 = 90^\circ$)

$$\begin{aligned} \boldsymbol{p}_1 &= e^{j\eta_1} \begin{bmatrix} \cos\theta_1 \cos\phi_1 \\ \cos\theta_1 \sin\phi_1 \\ -\sin\theta_1 \end{bmatrix} \\ \boldsymbol{p}_2 &= \begin{bmatrix} \cos\theta_2 \cos\phi_2 \sin\gamma_2 - \sin\phi_2 \cos\gamma_2 \\ \cos\theta_2 \sin\phi_2 \sin\gamma_2 + \cos\phi_2 \cos\gamma_2 \\ -\sin\theta_2 \sin\gamma_2 \end{bmatrix} \end{aligned} \tag{4.34}$$

For arbitrary given $\theta_1, \phi_1, \theta_2, \phi_2$ which satisfy the constraint (4.24), if $\boldsymbol{p}_1 // \boldsymbol{p}_2$, then

$$\begin{cases} \dfrac{\sin\theta_1}{\sin\theta_2 \sin\gamma_2} = \dfrac{\cos\theta_1 \cos\phi_1}{\cos\theta_2 \cos\phi_2 \sin\gamma_2 - \sin\phi_2 \cos\gamma_2} \\ \dfrac{\sin\theta_1}{\sin\theta_2 \sin\gamma_2} = \dfrac{\cos\theta_1 \sin\phi_1}{\cos\theta_2 \sin\phi_2 \sin\gamma_2 + \cos\phi_2 \cos\gamma_2} \end{cases} \tag{4.35}$$

leading to

$$\begin{cases} \sin\phi_2 = \cos\phi_2 \\ \sin\phi_2 = -\cos\phi_2 \end{cases} \tag{4.36}$$

which causes the contradiction. In this case, there is no ambiguity.

Case 2.4: $\gamma_1 = 0^\circ$ and $\gamma_2 = 0^\circ$.

$$\begin{aligned} \boldsymbol{p}_1 &= \begin{bmatrix} -\sin\phi_1 \\ \cos\phi_1 \\ 0 \end{bmatrix} \\ \boldsymbol{p}_2 &= \begin{bmatrix} -\sin\phi_2 \\ \cos\phi_2 \\ 0 \end{bmatrix} \end{aligned} \tag{4.37}$$

In this case, with $\phi_1 = \phi_2$, we have $\boldsymbol{p}_1 // \boldsymbol{p}_2$ for arbitrary η_1 and η_2. An example is $(30^\circ, 60^\circ, 0^\circ, 20^\circ)$ and $(30^\circ, 60^\circ, 0^\circ, 50^\circ)$.

Case 2.5: $\gamma_1 = 0^\circ$ and $\eta_2 = 0^\circ$. (same for $\eta_1 = 0^\circ$ and $\gamma_2 = 0^\circ$)

$$\begin{aligned} \boldsymbol{p}_1 &= \begin{bmatrix} -\sin\phi_1 \\ \cos\phi_1 \\ 0 \end{bmatrix} \\ \boldsymbol{p}_2 &= \begin{bmatrix} \cos\theta_2\cos\phi_2\sin\gamma_2 - \sin\phi_2\cos\gamma_2 \\ \cos\theta_2\sin\phi_2\sin\gamma_2 + \cos\phi_2\cos\gamma_2 \\ -\sin\theta_2\sin\gamma_2 \end{bmatrix} \end{aligned} \tag{4.38}$$

In this case, to satisfy the parallel condition, firstly θ_2 should be 0° and η_1 can be an arbitrary value. Further we have

$$\tan\gamma_2 = \frac{\cos\phi_1\sin\phi_2 - \sin\phi_1\cos\phi_2}{\cos\phi_1\cos\phi_2 + \sin\phi_1\sin\phi_2} \tag{4.39}$$

An example is $(30^\circ, 0^\circ, 0^\circ, 30^\circ)$ and $(0^\circ, 30^\circ, 30^\circ, 0^\circ)$.

Case 2.6: $\eta_1 = 0^\circ$ and $\eta_2 = 0^\circ$.

$$\begin{aligned} \boldsymbol{p}_1 &= \begin{bmatrix} \cos\theta_1\cos\phi_1\sin\gamma_1 - \sin\phi_1\cos\gamma_1 \\ \cos\theta_1\sin\phi_1\sin\gamma_1 + \cos\phi_1\cos\gamma_1 \\ -\sin\theta_1\sin\gamma_1 \end{bmatrix} \\ \boldsymbol{p}_2 &= \begin{bmatrix} \cos\theta_2\cos\phi_2\sin\gamma_2 - \sin\phi_2\cos\gamma_2 \\ \cos\theta_2\sin\phi_2\sin\gamma_2 + \cos\phi_2\cos\gamma_2 \\ -\sin\theta_2\sin\gamma_2 \end{bmatrix} \end{aligned} \tag{4.40}$$

In this case, due to the parallel condition, we know

$$\begin{cases} \dfrac{\sin\theta_1\sin\gamma_1}{\sin\theta_2\sin\gamma_2} = \dfrac{\cos\theta_1\cos\phi_1\sin\gamma_1 - \sin\phi_1\cos\gamma_1}{\cos\theta_2\cos\phi_2\sin\gamma_2 - \sin\phi_2\cos\gamma_2} \\ \dfrac{\sin\theta_1\sin\gamma_1}{\sin\theta_2\sin\gamma_2} = \dfrac{\cos\theta_1\sin\phi_1\sin\gamma_1 + \cos\phi_1\cos\gamma_1}{\cos\theta_2\sin\phi_2\sin\gamma_2 + \cos\phi_2\cos\gamma_2} \end{cases} \tag{4.41}$$

Each equations in (4.41) will produce a unique solution to $\tan\gamma_2$. Except that all the parameters $(\theta_1, \phi_1, \gamma_1) = (\theta_2, \phi_2, \gamma_2)$, there is no other solutions for γ_2 and therefore there is no ambiguity in this case.

4.3 Cramér-Rao Bound for Tripole Sensor Arrays

The Cramér-Rao bound (CRB) provides a lower bound on the variance of unbiased estimators. The estimator is said to be unbiased if the mean of the sampling distribution of a given parameter is equal to the parameter to be estimated. In paper [74], it has been proved that the MUSIC estimation errors are asymptotically joint Gaussian distributed with zero mean when the snapshot number is large, which means the MUSIC estimator is an unbiased estimator for the given parameters and CRB is suitable to evaluate the estimation performance. In the joint estimation problem, $(\theta, \phi, \gamma, \eta)$ are four unknown parameters. For convenience, we use $\boldsymbol{\alpha} = (\theta, \phi, \gamma, \eta)$ to denote the unknown parameters. With the N-element linear tripole sensor array, the probability density function for single received snapshot is given by [73]

$$p_x|(\boldsymbol{\alpha}) = \frac{1}{\det[\pi \boldsymbol{R}_{\boldsymbol{x}}(\boldsymbol{\alpha})]} e^{\{-[\boldsymbol{x}-\boldsymbol{m}(\boldsymbol{\alpha})]^H \boldsymbol{R}_{\boldsymbol{x}}^{-1}(\boldsymbol{\alpha})[\boldsymbol{x}-\boldsymbol{m}(\boldsymbol{\alpha})]\}} \tag{4.42}$$

where $\boldsymbol{R}_{\boldsymbol{x}}(\boldsymbol{\alpha})$ is the covariance matrix and $\boldsymbol{m}(\boldsymbol{\alpha})$ is the mean value of received vector data.

With K independent snapshots, the likelihood function can be denoted as the product of K single functions

$$\begin{aligned} p_{x_1,x_2,\ldots,x_K}|(\boldsymbol{\alpha}) = & \prod_{k=1}^{K} \frac{1}{\det[\pi \boldsymbol{R}_{\boldsymbol{x}}(\boldsymbol{\alpha})]} \\ & \times e^{\{-[\boldsymbol{x}_k-\boldsymbol{m}(\alpha)]^H \boldsymbol{R}_{\boldsymbol{x}}^{-1}(\alpha)[\boldsymbol{x}_k-\boldsymbol{m}(\boldsymbol{\alpha})]\}} \end{aligned} \tag{4.43}$$

The log-likelihood function is given by

$$\begin{aligned} L_x(\boldsymbol{\alpha}) = & \ln p_{x_1,x_2,\ldots,x_K}|(\boldsymbol{\alpha}) \\ = & -K \ln \det[\boldsymbol{R}_{\boldsymbol{x}}(\boldsymbol{\alpha})] - KN \ln \pi \\ & - \sum_{k=1}^{K} [\boldsymbol{x}_k - \boldsymbol{m}(\boldsymbol{\alpha})]^H \boldsymbol{R}_{\boldsymbol{x}}^{-1}(\boldsymbol{\alpha})[\boldsymbol{x}_k - \boldsymbol{m}(\boldsymbol{\alpha})] \end{aligned} \tag{4.44}$$

Considering the unconditional model, which means the source signals are random in

all realizations [85], , for one source the mean value and covariance matrix is given by

$$\begin{aligned} \boldsymbol{m}(\boldsymbol{\alpha}) &= \boldsymbol{0} \\ \boldsymbol{R_x} &= \sigma_s^2 \boldsymbol{v}\boldsymbol{v}^H + \sigma_n^2 \boldsymbol{I} \end{aligned} \tag{4.45}$$

where σ_s^2 is the power of source signal and σ_n^2 is the noise power.

The fisher information matrix can be denoted as:

$$\boldsymbol{F}(\boldsymbol{\alpha}) = \begin{bmatrix} F_{\theta,\theta} & F_{\theta,\phi} & F_{\theta,\gamma} & F_{\theta,\eta} \\ F_{\phi,\theta} & F_{\phi,\phi} & F_{\phi,\gamma} & F_{\phi,\eta} \\ F_{\gamma,\theta} & F_{\gamma,\phi} & F_{\gamma,\gamma} & F_{\gamma,\eta} \\ F_{\eta,\theta} & F_{\eta,\phi} & F_{\eta,\gamma} & F_{\eta,\eta} \end{bmatrix} \tag{4.46}$$

Each element in the matrix can be expressed as the product of derivatives of (5.70) with respect to the corresponding parameter [73]:

$$\begin{aligned} F_{\alpha_i,\alpha_j} =& \mathrm{tr}\{\boldsymbol{R_x}^{-1}(\boldsymbol{\alpha}) \frac{\partial \boldsymbol{R_x}(\boldsymbol{\alpha})}{\alpha_i} \boldsymbol{R_x}^{-1}(\boldsymbol{\alpha}) \frac{\partial \boldsymbol{R_x}(\boldsymbol{\alpha})}{\alpha_j}\} \\ &+ 2Re\{\frac{\partial \boldsymbol{m}^H(\boldsymbol{\alpha})}{\alpha_i} \boldsymbol{R_x}^{-1}(\boldsymbol{\alpha}) \frac{\partial \boldsymbol{m}(\boldsymbol{\alpha})}{\alpha_j}\} \end{aligned} \tag{4.47}$$

where the symbol tr{} means the trace of a matrix, $Re\{\}$ means the real part, and α_i, α_j denote two arbitrary parameters among $(\theta, \phi, \gamma, \eta)$.

With (5.74), (4.47) can be simplified to

$$F_{\alpha_i,\alpha_j} = \mathrm{tr}\{\boldsymbol{R_x}^{-1}(\boldsymbol{\alpha}) \frac{\partial \boldsymbol{R_x}(\boldsymbol{\alpha})}{\alpha_i} \boldsymbol{R_x}^{-1}(\boldsymbol{\alpha}) \frac{\partial \boldsymbol{R_x}(\boldsymbol{\alpha})}{\alpha_j}\} \tag{4.48}$$

The CRB matrix $\boldsymbol{C}(\boldsymbol{\alpha})$ is the inverse of fisher information matrix, i.e.

$$\boldsymbol{C}(\boldsymbol{\alpha}) = \boldsymbol{F}^{-1}(\boldsymbol{\alpha}) \tag{4.49}$$

Finally, the Cramér-Rao bounds for each estimated parameter are given by:

$$\begin{aligned} CRB(\theta) &= C_{\theta,\theta} = [\boldsymbol{F}^{-1}(\boldsymbol{\alpha})]_{1,1} \\ CRB(\phi) &= C_{\phi,\phi} = [\boldsymbol{F}^{-1}(\boldsymbol{\alpha})]_{2,2} \\ CRB(\gamma) &= C_{\gamma,\gamma} = [\boldsymbol{F}^{-1}(\boldsymbol{\alpha})]_{3,3} \\ CRB(\eta) &= C_{\eta,\eta} = [\boldsymbol{F}^{-1}(\boldsymbol{\alpha})]_{4,4} \end{aligned} \tag{4.50}$$

4.4 The Proposed Algorithm

In the following, the proposed low-complexity joint 4-D DOA and polarisation estimation algorithm for tripole sensor arrays is introduced based on a subspace approach.

Firstly, by applying eigenvalue decomposition (EVD), the covariance matrix $\boldsymbol{R}$ can be decomposed into

$$\boldsymbol{R} = \boldsymbol{R}_s + \boldsymbol{R}_n = \sum_{k=1}^{3N} \lambda_k \boldsymbol{u}_k \boldsymbol{u}_k^H \tag{4.51}$$

where $\boldsymbol{u}_k$ is the k-th eigenvector and λ_k is the corresponding eigenvalue (in descending order). Furthermore, we can rewrite (4.51) into

$$\boldsymbol{R} = \boldsymbol{U}_s \boldsymbol{\Lambda}_{\mathbf{s}} \boldsymbol{U}_s^H + \boldsymbol{U}_n \boldsymbol{\Lambda}_n \boldsymbol{U}_n^H \tag{4.52}$$

where $\boldsymbol{U}_s = [\boldsymbol{u}_1, \boldsymbol{u}_2, \cdots, \boldsymbol{u}_M]$ and $\boldsymbol{U}_n = [\boldsymbol{u}_{M+1}, \boldsymbol{u}_{M+2}, \cdots, \boldsymbol{u}_{3N}]$ are the eigenvectors of the signal subspace and noise subspace, respectively. $\boldsymbol{\Lambda}_{\mathbf{s}}$ and $\boldsymbol{\Lambda}_n$ are diagonal matrices holding the corresponding eigenvalues λ_k. As the rank of the noise subspace cannot be less than 1, the DOF (degree of freedom) of the algorithm is $3N - 1$.

Clearly, the joint steering vector $\boldsymbol{v}_m$ is orthogonal to the noise subspace $\boldsymbol{U}_n$, i.e.

$$\boldsymbol{U}_n^H \boldsymbol{v}_m = \boldsymbol{0} \tag{4.53}$$

or

$$\boldsymbol{v}_m^H \boldsymbol{U}_n \boldsymbol{U}_n^H \boldsymbol{v}_m = 0 \tag{4.54}$$

As a result, to find the DOA and polarisation parameters $(\theta_m, \phi_m, \gamma_m, \eta_m)$ of the m-th signal, we construction the following function

$$F(\theta, \phi, \gamma, \eta) = \frac{1}{\boldsymbol{v}^H \boldsymbol{U}_n \boldsymbol{U}_n^H \boldsymbol{v}} \tag{4.55}$$

The peaks in (4.55) indicate the DOA and polarisation information $(\theta, \phi, \gamma, \eta)$ for impinging signals.

The above MUSIC-type algorithm is based on direct 4-D peak search with an extremely large computational complexity. In the following, we transform the 4-D search process into two 2-D searches, significantly reducing the complexity of the solution.

First, we separate $\boldsymbol{v}_m$ into two components: one with DOA information (θ, ϕ) only, while the other only contains the polarisation information (γ, η). In this way, (4.53) can be changed to

$$\begin{aligned} \mathbf{0} &= \boldsymbol{U}_n^H[\boldsymbol{a}_m \otimes (\boldsymbol{\Omega}_m \boldsymbol{g}_m)] \\ &= \boldsymbol{U}_n^H[(\boldsymbol{a}_m \otimes \boldsymbol{\Omega}_m)\boldsymbol{g}_m] \\ &= [\boldsymbol{U}_n^H \boldsymbol{b}_m]\boldsymbol{g}_m \end{aligned} \tag{4.56}$$

where $\boldsymbol{b}_m$ is the Kronecker product of $\boldsymbol{a}_m$ and $\boldsymbol{\Omega}_m$.

Note that $\boldsymbol{U}_n^H \boldsymbol{b}_m$ is a $(3N - M) \times 2$ vector and $\boldsymbol{g}_m$ is a 2×1 vector. (4.56) shows a linear relationship between the columns in $\boldsymbol{U}_n^H \boldsymbol{b}_m$, which means the vector has a column rank less than 2. Multiplied by its Hermitian transpose on the right, the new 2×2 product matrix cannot have a full rank, with its determinant equal to zero. Here we use $\mathbf{det}\{\}$ to denote the determinant of a matrix. Then, we have

$$\mathbf{det}\{\boldsymbol{b}_m^H \boldsymbol{U}_n \boldsymbol{U}_n^H \boldsymbol{b}_m\} = 0 \tag{4.57}$$

We can see that $\boldsymbol{b}_m$ is dependent on the parameters (θ, ϕ) only. As a result, a new estimator can be established corresponding to θ and ϕ as [86]

$$f(\theta, \phi) = \frac{1}{\mathbf{det}\{\boldsymbol{b}^H \boldsymbol{U}_n \boldsymbol{U}_n^H \boldsymbol{b}\}} \tag{4.58}$$

The new estimator first performs a 2-D peak search over θ and ϕ. After locating θ and ϕ, the polarisation parameters γ and η can be obtained by another 2-D search in the following

$$f(\gamma, \eta) = \frac{1}{\boldsymbol{g}^H \boldsymbol{b}^H \boldsymbol{U}_n \boldsymbol{U}_n^H \boldsymbol{b}\boldsymbol{g}} \tag{4.59}$$

The following is a summary of the proposed algorithm:

- Calculate the estimated covariance matrix $\hat{\boldsymbol{R}}$ from the received signals.
- Calculate the noise space $\boldsymbol{U}_n$ by applying the eigenvalue decomposition on $\hat{\boldsymbol{R}}$. The last $3N - M$ eigenvalues and the corresponding eigenvectors form the noise space.
- Use the 2-D estimator (4.58) to locate the DOA parameters θ and ϕ.
- Use the 2-D estimator (4.59) to locate the corresponding polarisation parameters γ and η.

4.5 Simulation Results

In this section, simulation results are presented to demonstrate the ambiguity issues we discussed earlier and the performance of the proposed algorithm.

4.5.1 Ambiguity phenomenon

Assuming one source signal from $(\theta, \phi, \gamma, \eta) = (30^\circ, 80^\circ, 20^\circ, 50^\circ)$ impinges on a uniform linear crossed-dipole array and a uniform linear tripole array respectively. Both arrays have the same senor number $N = 5$ and the inter-element space is set to $d = \lambda/2$.

Fig. 4.2 and Fig. 4.3 present the DOA estimation results for these two arrays, respectively. Apparently, the tripole sensor array gives a unique peak point at the source direction while the crossed-dipole senor array shows a peak line due to the ambiguity problem and there is no way to identify the real direction of the signal.

4.5.2 RMSE results

Now we study the performance of the proposed algorithm based on tripole sensor arrays. There are two source signals from $(\theta, \phi, \gamma, \eta) = (10^\circ, 20^\circ, 15^\circ, 30^\circ)$ and $(\theta, \phi, \gamma, \eta) = (60^\circ, 70^\circ, 60^\circ, 80^\circ)$. The tripole sensor number is set to $N = 4$ and the number of snapshots for each simulation is $K = 1000$. The root mean square error (RMSE) results of the estimated parameters by 200 Monte-Carlo trials are shown in Fig. 4.4-4.7, where we can

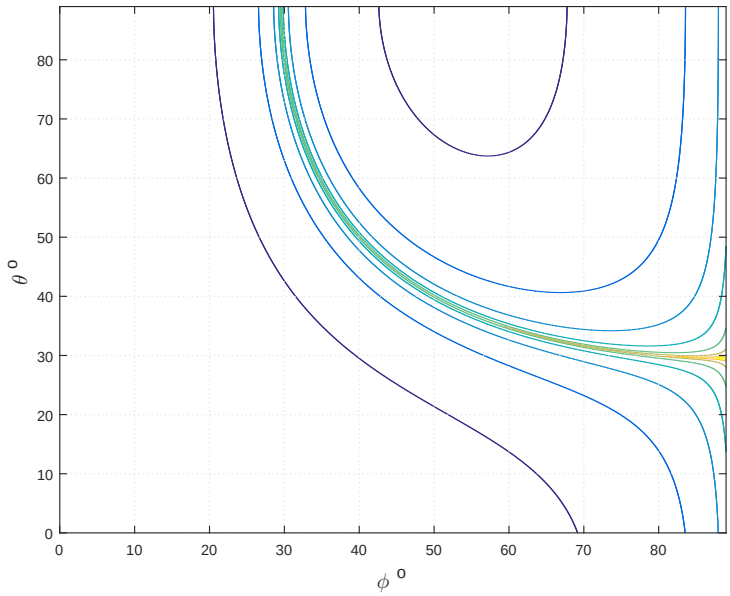

Figure 4.2: DOA estimation result using the linear crossed-dipole sensor array.

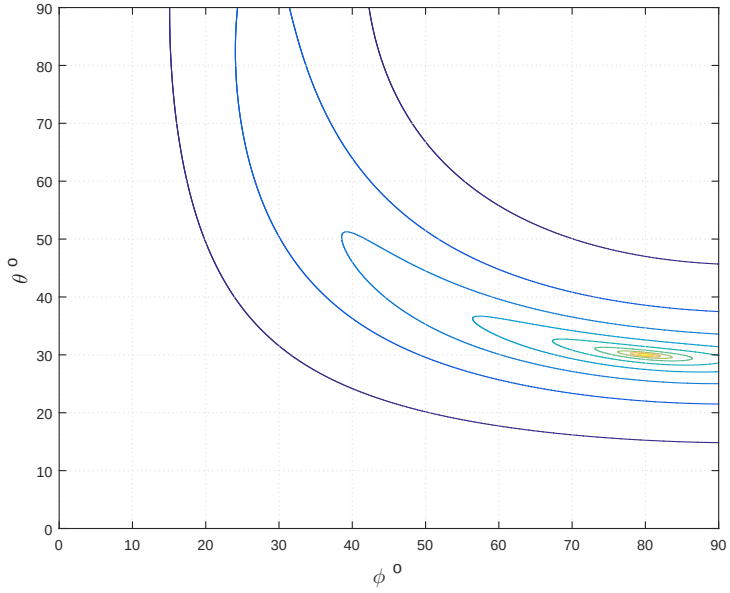

Figure 4.3: DOA estimation result using the linear tripole sensor array.

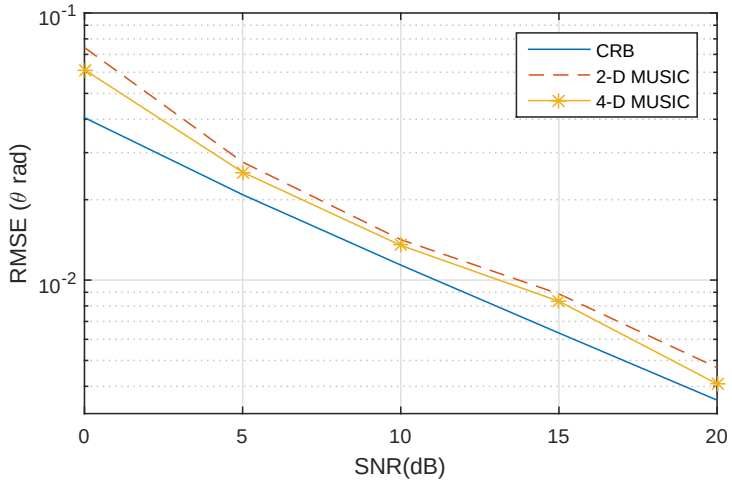

Figure 4.4: RMSE of θ.

see that with the increase of SNR, the RMSE level decreases consistently. The accuracy of the 4-D MUSIC using (4.55) is always better than the proposed 2-D MUSIC algorithm for any parameters at the cost of a much higher level of computation complexity. The performance of both algorithms are close to the CRB.

4.5.3 Linear tripole and planar crossed-dipole array

Since a planar crossed-dipole array can also be used to estimate the four parameters of an impinging signal, it would be interesting to know that given the same number of dipoles, which one is more effective for 4-D parameter estimation, the linear tripole array or the planar crossed-dipole array. To find out, in this part, we consider a 4×1 linear tripole array and a 2×3 planar crossed-dipole array both of which have the same number of dipoles or DOFs. We compare their estimation accuracy using the proposed 2-D MUSIC algorithm. All the other conditions are the same as in 4.5.2.

Fig. 5.10 shows the RMSE results for the first signal's azimuth angle. It can be seen that the planar array has given a higher estimating accuracy and its CRB is much lower than the linear tripole array, which means that the compact structure of the linear tripole sensor array is achieved at the cost of estimation accuracy. However, it's hard to say

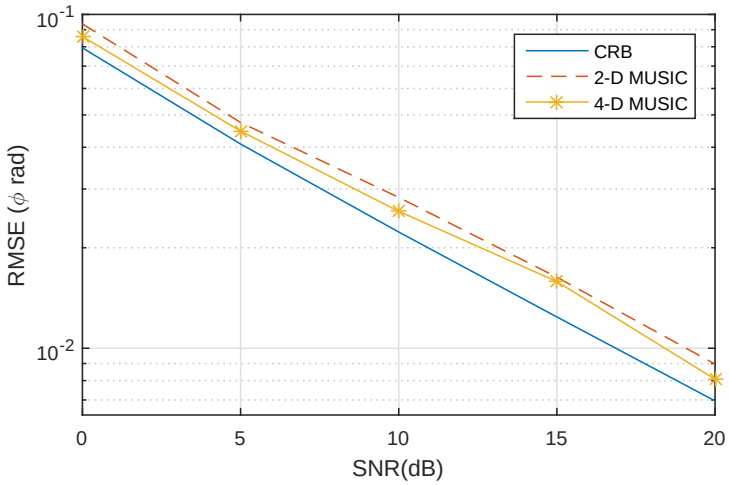

Figure 4.5: RMSE of ϕ.

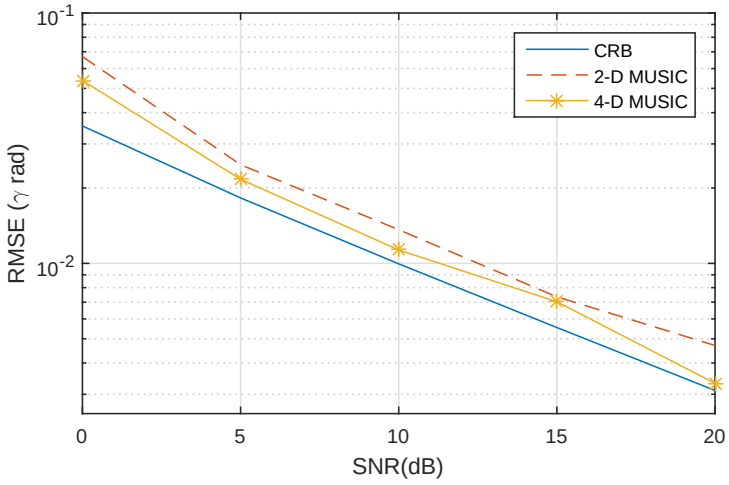

Figure 4.6: RMSE of γ.

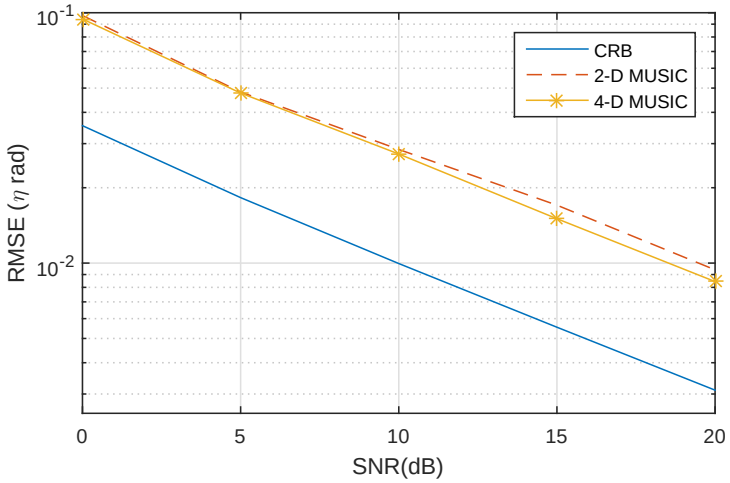

Figure 4.7: RMSE of η.

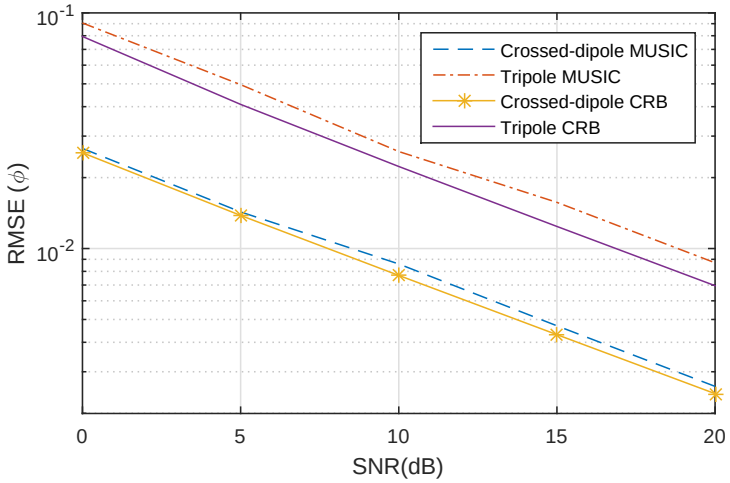

Figure 4.8: RMSE of crossed-dipole and tripole sensor array.

that the planar crossed-dipole array outperforms the linear triple array totally. In some special scenarios, if the sensor array has to be placed in a narrow area, for example, on the surface of an airplane. The planar array will result in the short inter-space in one direction, which may cause other problems like mutual coupling between sensors. While linear array can avoid this and ensure enough distance between sensors.

4.6 Summary

With a detailed analysis and proof, it has been shown that due to inherent limitation of the linear crossed-dipole structure, it cannot uniquely identify the four parameters associated with impinging signals. In order to simultaneously estimate both the 2-D DOA and 2-D polarisation parameters of the impinging signals, we could increase the dimension of the array and construct a planar crossed-dipole array. To avoid this and have a compact structure, a linear tripole array has been employed instead. It has been proved and also shown that such a structure can estimate the 2-D DOA and 2-D polarisation information effectively except for some very special cases. Moreover, a dimension-reduction based MUSIC algorithm was developed so that the 4-D estimation problem can be simplified to two separate 2-D estimation problems, significantly reducing the computational complexity of the solution. However, the dimension-reduction also brings the problem of less accuracy. Since both the planar crossed-dipole array and the linear tripole array can be used to effectively estimate the four parameters of an impinging signal, a brief comparison between them was also carried out and it was shown that given the same number of dipoles, the planar structure has a better performance, although this is achieved at the cost of increased physical size.

Chapter 5

Direction Finding for Signals with Mixed Signal Transmission Models

5.1 Introduction

Direction of arrival estimation has been widely studied in recent years [63, 64, 87, 88], and many algorithms have been introduced to solve the DOA estimation problem, such as MUSIC [26], estimation of signal parameters via rotational invariance techniques (ESPRIT) [27] and those based on sparsity or compressive sensing (CS) [89, 90]. In its early time, most research on DOA estimation was based on omnidirectional antennas, ignoring the polarisation information of impinging signals. To consider the polarisation information, electromagnetic vector sensor arrays were proposed to jointly estimate the DOA and polarisation information [16, 34, 43, 91]. The MUSIC, ESPRIT and CS-based algorithms can be extended to solve the joint DOA and polarisation estimation problem [61, 62, 67, 72, 92–95]. However, in their models, for each direction, it is assumed either explicitly or implicitly that there is only one signal impinging upon the array; in other words, each source only emits one single signal with specific direction and polarisation and we refer to such system as a single signal transmission (SST) system.

To make full use of the degree of freedom provided by a vector sensor array, two

separate signals could be transmitted simultaneously from each source, and this is refer to the dual signal transmission (DST) model [96]. For a DST signal, the two sub-signals have the same DOA but different polarisations. One DST example is to use two orthogonal linearly polarized signals with amplitude or phase modulation [97, 98]. However, there has rarely been any research reported on estimating the DOAs of DST signals. Instinctively, we could consider a DST signal as two independent SST signals and estimate their DOAs one by one. However, as we will see later, a direct application of the traditional DOA estimation methods such as the subspace-based ones may not work expected for DST signals and a new approach is needed.

In this thesis, based on a uniform linear tripole sensor array, we first try to extend the classic MUSIC algorithm straightforwardly to the 4-D case to find the parameters of a mixture of impinging SST and DST signals. As analysed later, due to inherent physical property of signal polarisation and array structure, we can only find the DOA and polarisation parameters of SST signals and for the DST signals, it fails due to an ambiguity problem with their estimation. The ambiguity problem associated with the polarisation parameters of DST signals cannot be solved by any estimator due to limitation of the degrees of freedom available in the polarisation domain. However, it is possible to obtain the DOA information of DST signals. As a solution and also to reduce the complexity of the 4-D search process of the extended MUSIC algorithm and also exploit the additional information provided by DST signals, i.e. the two sub-signals of each DST signal share the same DOA, a two-step algorithm is proposed. In this solution, the DOA and polarisation information of SST signals are found first by a rank-reduction algorithm (referred to as the SST estimator) and then the DOA information of the DST signals is estimated by a specifically designed estimator (referred to as the DST estimator). Furthermore, a general estimator (referred to as the MST estimator) is proposed which can obtain the DOA parameters of the SST and DST signals in one single step, while the polarisation information of SST signals can be obtained by a separate 2-D search if needed. Moreover, the CRB (Cramér-Rao Bound) is derived to evaluate the performance of the

proposed estimation algorithms. As demonstrated by simulation results, for SST signals, the two proposed estimators (the two-step estimator and the general estimator) have a similar performance, while the general estimator has a higher accuracy in estimating the direction of DST signals.

5.2 Signal Models

In our mixed signal transmission model, there are M_1 SST and M_2 DST narrowband polarized sources impinging on a uniform linear array with N tripole sensors from the far field as shown in Fig. 5.1. Each SST source emits only one signal $s_m(t), m = 1, 2, \cdots, M_1$, and each DST source emits two sub-signals $s_{M_1+2m-1}(t)$ and $s_{M_1+2m}(t), m = 1, 2, \cdots, M_2$, with the same elevation-azimuth angle (θ, ϕ) but different polarisation (γ, η), where γ, η denote the polarisation auxiliary angle and the polarisation phase difference, respectively. For convenience, the parameters of the DST signal s_{M_1+2m-1} and s_{M_1+2m} are denoted by $(\theta_{M_1+2m-1}, \phi_{M_1+2m-1}, \gamma_{M_1+2m-1}, \eta_{M_1+2m-1})$ and $(\theta_{M_1+2m}, \phi_{M_1+2m}, \gamma_{M_1+2m}, \eta_{M_1+2m})$, respectively. Note that

$$\begin{cases} \theta_{M_1+2m-1} = \theta_{M_1+2m} \\ \phi_{M_1+2m-1} = \phi_{M_1+2m} \end{cases} \tag{5.1}$$

In discrete form, the received SST signals of a single tripole sensor at the k-th time instant is denoted by a 3×1 vector $\boldsymbol{x}_s[k]$ (noise-free)

$$\boldsymbol{x}_s[k] = \sum_{m=1}^{M_1} \boldsymbol{p}_m s_m[k] \tag{5.2}$$

where $\boldsymbol{p}_m$ is the SST angular-polarisation vector given by

$$\begin{aligned} \boldsymbol{p}_m &= \begin{bmatrix} \cos\theta_m \cos\phi_m & -\sin\phi_m \\ \cos\theta_m \sin\phi_m & \cos\phi_m \\ -\sin\theta_m & 0 \end{bmatrix} \begin{bmatrix} \sin\gamma_m e^{j\eta_m} \\ \cos\gamma_m \end{bmatrix} \\ &= \boldsymbol{\Omega}_m \cdot \boldsymbol{g}_m \end{aligned} \tag{5.3}$$

In the above equation, $\boldsymbol{\Omega}_m$ denotes the angular matrix associated with DOA parameters θ and ϕ, and $\boldsymbol{g}_m$ is the polarisation vector including polarisation parameters γ and η, given by

$$\boldsymbol{\Omega}_m = \begin{bmatrix} \cos\theta_m \cos\phi_m & -\sin\phi_m \\ \cos\theta_m \sin\phi_m & \cos\phi_m \\ -\sin\theta_m & 0 \end{bmatrix} \tag{5.4}$$

$$\boldsymbol{g}_m = \begin{bmatrix} \sin\gamma_m e^{j\eta_m} \\ \cos\gamma_m \end{bmatrix} \tag{5.5}$$

The DST signals collected by a single tripole sensor can be considered as the sum of all $2M_2$ sub-signals, where each sub-signal can be viewed as a SST signal. Hence, the received DST signals are in the form

$$\boldsymbol{x}_d[k] = \sum_{m=M_1}^{M_1+2M_2} \boldsymbol{p}_m s_m[k] \tag{5.6}$$

Considering a pair of sub-signals as a single composite DST signal, we can use a 2×1 vector $\boldsymbol{s}_m$ to denote the m-th DST signal corresponding to the pair of sub-signals s_{M_1+2m-1} and s_{M_1+2m}, which is defined by

$$\boldsymbol{s}_m[k] = \begin{bmatrix} s_{M_1+2m-1}[k] \\ s_{M_1+2m}[k] \end{bmatrix} \tag{5.7}$$

Then, (5.6) can be transformed to

$$\begin{aligned} \boldsymbol{x}_d[k] &= \sum_{m=1}^{M_2} [\boldsymbol{p}_{M_1+2m-1} \quad \boldsymbol{p}_{M_1+2m-1}] \begin{bmatrix} s_{M_1+2m-1}[k] \\ s_{M_1+2m}[k] \end{bmatrix} \\ &= \sum_{m=1}^{M_2} \boldsymbol{P}_m \boldsymbol{s}_m[k] \end{aligned} \tag{5.8}$$

where $\boldsymbol{P}_m$ is the angular-polarisation matrix for DST signals, with

$$\begin{aligned} \boldsymbol{P}_m &= [\boldsymbol{p}_{M_1+2m-1} \quad \boldsymbol{p}_{M_1+2m-1}] \\ &= [\boldsymbol{\Omega}_{M_1+2m-1}\boldsymbol{g}_{M_1+2m-1} \quad \boldsymbol{\Omega}_{M_1+2m}\boldsymbol{g}_{M_1+2m}] \end{aligned} \tag{5.9}$$

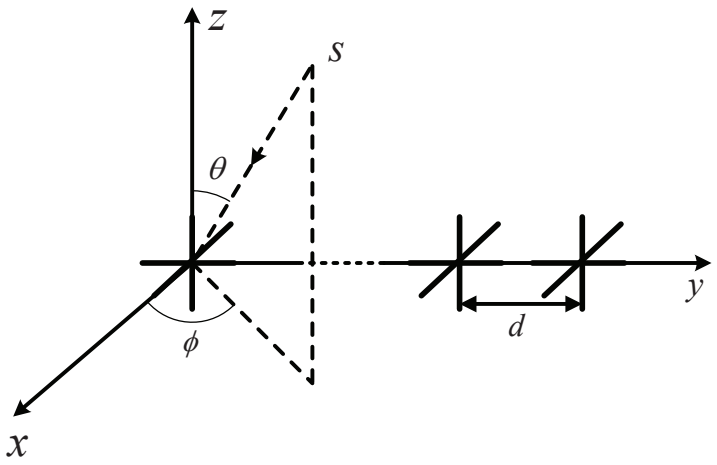

Figure 5.1: Geometry of a uniform linear tripole array, where a signal arrives from elevation angle θ and azimuth angle ϕ.

Note that the two sub-signals of the same DST signal share the same angular matrix, and here we use $\boldsymbol{\Xi}_m$ represent the common angular matrix of the m-th DST signal, i.e.

$$\boldsymbol{\Xi}_m = \boldsymbol{\Omega}_{M_1+2m-1} = \boldsymbol{\Omega}_{M_1+2m} \tag{5.10}$$

We use $\boldsymbol{G}_m$ to denote the polarisation matrix of the m-th DST signal, defined as

$$\boldsymbol{G}_m = [\boldsymbol{g}_{M_1+2m-1} \quad \boldsymbol{g}_{M_1+2m}] \tag{5.11}$$

Then $\boldsymbol{P}_m$ is the product of $\boldsymbol{\Xi}_m$ and $\boldsymbol{G}_m$,

$$\boldsymbol{P}_m = \boldsymbol{\Xi}_m \boldsymbol{G}_m \tag{5.12}$$

The total received signal $\boldsymbol{x}[k]$ is the sum of SST and DST signals, which is given by

$$\begin{aligned} \boldsymbol{x}[k] &= \boldsymbol{x}_s[k] + \boldsymbol{x}_d[k] \\ &= \sum_{m=1}^{M_1} \boldsymbol{p}_m s_m[k] + \sum_{m=1}^{M_2} \boldsymbol{P}_m \boldsymbol{s}_m[k] \end{aligned} \tag{5.13}$$

Now we consider the whole array system. The phase delay of source signals collected by different tripole sensors is defined as the steering vector, $\boldsymbol{a}_m$, given by

$$\boldsymbol{a}_m = [1, e^{-j\tau \sin\theta_m \sin\phi_m}, ..., e^{-j(N-1)\tau \sin\theta_m \sin\phi_m}]^T \tag{5.14}$$

where τ is the phase shift between two adjacent sensors in the array

$$\tau = \frac{2\pi d}{\lambda} \tag{5.15}$$

with λ being the wavelength of the narrowband signals and d is the spacing between adjacent tripole sensors.

Firstly, consider the each-sub signal as a separate SST signal. With the Gaussian white noise $\boldsymbol{n}[k]$ of variance σ_n^2, the array output snapshot at the k-th time instant $\boldsymbol{y}[k]$ is given by [72]

$$\boldsymbol{y}[k] = \sum_{m=1}^{M_1+2M_2} \boldsymbol{a}_m \otimes \boldsymbol{p}_m \cdot s_m[k] + \boldsymbol{n}[k] \tag{5.16}$$

Consider every pair of sub-signals as a DST signal, (5.16) will be transformed to

$$\begin{aligned} \boldsymbol{y}[k] &= \sum_{m=1}^{M_1} \boldsymbol{a}_m \otimes \boldsymbol{p}_m \cdot s_m[k] \\ &+ \sum_{m=1}^{M_2} \boldsymbol{a}_{M_1+2m-1} \otimes \boldsymbol{P}_m \cdot \boldsymbol{s}_m[k] + \boldsymbol{n}[k] \end{aligned} \tag{5.17}$$

In the DST signal part, as each pair of sub-signals come from the same direction, the two steering vectors are equal to each other, i.e.

$$\boldsymbol{a}_{M_1+2m-1} = \boldsymbol{a}_{M_1+2m} \tag{5.18}$$

Therefore, in (5.17), $\boldsymbol{a}_{M_1+2m-1}$ can be replaced by $\boldsymbol{a}_{M_1+2m}$, too.

To further simplify (5.17), $\boldsymbol{q}_m$ is used to denote the direction-polarisation joint steering vector for SST signals, defined as

$$\boldsymbol{q}_m = \boldsymbol{a}_m \otimes \boldsymbol{p}_m \tag{5.19}$$

and $\boldsymbol{Q}_m$ is the DST joint steering matrix, given by

$$\begin{aligned} \boldsymbol{Q}_m &= \boldsymbol{a}_{M_1+2m-1} \otimes \boldsymbol{P}_m \\ &= \boldsymbol{a}_{M_1+2m-1} \otimes [\boldsymbol{p}_{M_1+2m-1} \quad \boldsymbol{p}_{M_1+2m-1}] \\ &= [\boldsymbol{q}_{M_1+2m-1} \quad \boldsymbol{q}_{M_1+2m}] \end{aligned} \tag{5.20}$$

With the above notation, (5.17) is further changed to

$$\boldsymbol{y}[k] \;=\; \sum_{m=1}^{M_1} \boldsymbol{q}_m \cdot s_m[k] + \sum_{m=1}^{M_2} \boldsymbol{Q}_m \cdot \boldsymbol{s}_m[k] + \boldsymbol{n}[k] \tag{5.21}$$

5.3 Proposed Estimators

As mentioned in Introduction, the traditional subspace-based DOA estimation algorithms could be used to find the DOA of both SST and DST signals. Therefore we will try to extend the classic MUSIC algorithm to the 4-D case to show its limitations for a mixture of impinging SST and DST signals. To reduce the complexity of the 4-D search process and also exploit the additional information carried by DST signals, a two-step algorithm is proposed, consisting of two estimators, one for SST signals and one for DST signals. The one-step general MST estimator is proposed after that with a detailed proof for its working.

Before introducing the proposed estimators in detail, we first provide some basic concepts for subspace based DOA estimation method as the starting point. The covariance matrix of the received signals is given by

$$\boldsymbol{R} \;=\; E\{\boldsymbol{y}[k]\boldsymbol{y}[k]^H\} \tag{5.22}$$

which can be estimated by a finite number of snapshots as follows

$$\hat{\boldsymbol{R}} \approx \frac{1}{K}\sum_{k=1}^{K} \boldsymbol{y}[k]\boldsymbol{y}[k]^H \tag{5.23}$$

where K is the number of snapshots.

For the general MST model, assume $M_1 + 2M_2 < 3N$. After eigendecomposition, the covariance matrix can be expressed by

$$\boldsymbol{R} = \sum_{n=1}^{3N} \lambda_n \boldsymbol{u}_n \boldsymbol{u}_n^H \tag{5.24}$$

where λ_n is the n-th eigenvalue and $\boldsymbol{u}_n$ is the associated eigenvector. After sorting the $3N$ eigenvalues in descending order, the eigenvectors $\boldsymbol{u}_1, \boldsymbol{u}_2, ..., \boldsymbol{u}_{M_1+2M_2}$ form the signal

subspace, while $\boldsymbol{u}_{M_1+2M_2+1}, \boldsymbol{u}_{M_1+2M_2+2}, ..., \boldsymbol{u}_{3N}$ form the noise subspace. We use $\boldsymbol{U}_s$ and $\boldsymbol{U}_n$ to denote these two subspaces

$$\begin{aligned} \boldsymbol{U}_s &= [\boldsymbol{u}_1, \boldsymbol{u}_2, \cdots, \boldsymbol{u}_{M_1+2M_2}] \\ \boldsymbol{U}_n &= [\boldsymbol{u}_{M_1+2M_2+1}, \boldsymbol{u}_{M_1+2M_2+2}, \cdots, \boldsymbol{u}_{3N}] \end{aligned} \tag{5.25}$$

Clearly, both the SST joint steering vector $\boldsymbol{q}_m$ and the DST joint steering matrix $\boldsymbol{Q}_m$ are orthogonal to the noise subspace $\boldsymbol{U}_n$.

5.3.1 Extension of the traditional MUSIC estimator to 4-D

A traditional DOA estimator considers all incoming signals as seperate SST signals, i.e. the given M_1 SST signals and M_2 DST signals will be considered as $M_1 + 2M_2$ SST signals in the algorithm. The joint steering vector $\boldsymbol{q}_m$ is already introduced in (5.19). After applying eigenvalue decomposition, the signal subspace and noise subspace can be calculated by (5.24), (5.25). Since the joint steering vectors $\boldsymbol{q}_m$ are orthogonal to the noise subspace, then

$$\boldsymbol{U}_n^H \boldsymbol{q}_m = \boldsymbol{0} \tag{5.26}$$

By (5.3) and (5.19), it can be deduced that

$$\boldsymbol{q}_m = \boldsymbol{a}_m \otimes \boldsymbol{p}_m = \boldsymbol{a}_m \otimes (\boldsymbol{\Omega}_m \boldsymbol{g}_m) \tag{5.27}$$

Then (5.26) can be further transformed to

$$\boldsymbol{U}_n^H \boldsymbol{a}_m \otimes (\boldsymbol{\Omega}_m \boldsymbol{g}_m) = \boldsymbol{0} \tag{5.28}$$

The DOA and polarisation parameters will be estimated by finding the peaks of the following cost function through a 4-D search.

$$F(\theta, \phi, \gamma, \eta) = \frac{1}{\boldsymbol{q}_m^H \boldsymbol{U}_n \boldsymbol{U}_n^H \boldsymbol{q}_m} \tag{5.29}$$

However, as shown in the following analysis, there is an ambiguity problem with both DOA and polarisation of DST signals. As a result, their DOA and polarisation information

can not be obtained by the subspace based method. In the following, we give a detailed explanation to the ambiguity problem.

Firstly, we give the explanation to the ambiguity problem associated with the polarisation parameters. When the estimator is applied to a DST signal, the noise subspace will be orthogonal to the joint steering vectors of both sub-signals. Using $\boldsymbol{q}_1$ and $\boldsymbol{q}_2$ to denote the two different joint steering vectors, we have

$$\begin{cases} \boldsymbol{U}_n^H \boldsymbol{q}_1 = \boldsymbol{0} \\ \boldsymbol{U}_n^H \boldsymbol{q}_2 = \boldsymbol{0} \end{cases} \tag{5.30}$$

This can be further derived that

$$\begin{cases} \boldsymbol{U}_n^H \boldsymbol{a}_1 \otimes \boldsymbol{p}_1 = \boldsymbol{0} \\ \boldsymbol{U}_n^H \boldsymbol{a}_2 \otimes \boldsymbol{p}_2 = \boldsymbol{0} \end{cases}$$

$$\Rightarrow \begin{cases} \boldsymbol{U}_n^H \boldsymbol{a}_1 \otimes (\boldsymbol{\Omega}_1 \boldsymbol{g}_1) = \boldsymbol{0} \\ \boldsymbol{U}_n^H \boldsymbol{a}_2 \otimes (\boldsymbol{\Omega}_2 \boldsymbol{g}_2) = \boldsymbol{0} \end{cases} \tag{5.31}$$

Note $\boldsymbol{a}_1$ is an $N \times 1$ vectors, and $a_{11}, a_{12}, ..., a_{1N}$ denote the elements in $\boldsymbol{a}_1$. Then we have

$$\begin{aligned} \boldsymbol{a}_1 \otimes (\boldsymbol{\Omega}_1 \boldsymbol{g}_1) &= [a_{11}\boldsymbol{\Omega}_1 \boldsymbol{g}_1, a_{12}\boldsymbol{\Omega}_1 \boldsymbol{g}_1, ..., a_{1N}\boldsymbol{\Omega}_1 \boldsymbol{g}_1]^T \\ &= [a_{11}\boldsymbol{\Omega}_1, a_{12}\boldsymbol{\Omega}_1, ..., a_{1N}\boldsymbol{\Omega}_1]^T \boldsymbol{g}_1 \\ &= (\boldsymbol{a}_1 \otimes \boldsymbol{\Omega}_1)\boldsymbol{g}_1 \end{aligned} \tag{5.32}$$

Similar to $\boldsymbol{a}_2$, we have

$$\boldsymbol{a}_2 \otimes (\boldsymbol{\Omega}_2 \boldsymbol{g}_2) = (\boldsymbol{a}_2 \otimes \boldsymbol{\Omega}_2)\boldsymbol{g}_2 \tag{5.33}$$

Then, (5.31) can be changed to

$$\begin{cases} \boldsymbol{U}_n^H (\boldsymbol{a}_1 \otimes \boldsymbol{\Omega}_1)\boldsymbol{g}_1 = \boldsymbol{0} \\ \boldsymbol{U}_n^H (\boldsymbol{a}_2 \otimes \boldsymbol{\Omega}_2)\boldsymbol{g}_2 = \boldsymbol{0} \end{cases} \tag{5.34}$$

Since the two sub-signals come from the same direction, we have

$$\boldsymbol{a}_1 \otimes \boldsymbol{\Omega}_1 = \boldsymbol{a}_2 \otimes \boldsymbol{\Omega}_2 \tag{5.35}$$

On the other hand, as the two sub-signals have different polarisation, the vector $\boldsymbol{g}_1$ is not in parallel with $\boldsymbol{g}_2$. Hence, it must satisfy that

$$\boldsymbol{U}_n^H(\boldsymbol{a}_1 \otimes \boldsymbol{\Omega}_1) = \boldsymbol{0} \tag{5.36}$$

Then, given a third arbitrary polarisation vector $\boldsymbol{g}_3$, different from $\boldsymbol{g}_1$ and $\boldsymbol{g}_2$, we have

$$\boldsymbol{U}_n^H(\boldsymbol{a}_1 \otimes \boldsymbol{\Omega}_1) \cdot \boldsymbol{g}_3 = \boldsymbol{0} \tag{5.37}$$

As a result, $F(\theta_1, \phi_1, \gamma_3, \eta_3)$ will be recognised as a peak in the spectrum and wrongly identified as the parameters of a non-existing source. This means the algorithm fails when trying to estimate the polarisation of DST signals.

Next, we give an analysis to the ambiguity problem associated with the DOA of DST signals.

Given $\boldsymbol{q}_1$ and $\boldsymbol{q}_2$ as the joint steering vectors of the two sub-signals. By (5.32), (5.33) and (5.35), we have

$$\begin{cases} \boldsymbol{q}_1 = (\boldsymbol{a}_1 \otimes \boldsymbol{\Omega}_1)\boldsymbol{g}_1 \\ \boldsymbol{q}_2 = (\boldsymbol{a}_1 \otimes \boldsymbol{\Omega}_1)\boldsymbol{g}_2 \end{cases} \tag{5.38}$$

which can be further derived to

$$\begin{cases} \boldsymbol{q}_1 = (\boldsymbol{a}_1 \otimes \boldsymbol{I}_3)\boldsymbol{\Omega}_1\boldsymbol{g}_1 \\ \boldsymbol{q}_2 = (\boldsymbol{a}_1 \otimes \boldsymbol{I}_3)\boldsymbol{\Omega}_1\boldsymbol{g}_2 \end{cases} \tag{5.39}$$

where $\boldsymbol{I}_3$ is a 3×3 identity matrix. Consider a non-existing source s_3 with its DOA angles satisfying

$$\begin{cases} \sin\theta_3 \sin\phi_3 = \sin\theta_1 \sin\phi_1 \\ \sin\theta_3 \neq \sin\theta_1 \\ \sin\phi_3 \neq \sin\phi_1 \end{cases} \tag{5.40}$$

Then, it can be obtained that

$$\boldsymbol{a}_1 = \boldsymbol{a}_3 \tag{5.41}$$

To ensure $\boldsymbol{q}_3$ is the linear combination of $\boldsymbol{q}_1$ and $\boldsymbol{q}_2$, which can be denoted by

$$\boldsymbol{q}_3 = k_1 \boldsymbol{q}_1 + k_2 \boldsymbol{q}_2 \tag{5.42}$$

where k_1, k_2 are scalars (not all zero). Since the three steering vectors have the same component $\boldsymbol{a}_1 \otimes \boldsymbol{I}_3$, (5.42) can be transformed to

$$\boldsymbol{\Omega}_3 \boldsymbol{g}_3 = k_1 \boldsymbol{\Omega}_1 \boldsymbol{g}_1 + k_2 \boldsymbol{\Omega}_1 \boldsymbol{g}_2 \tag{5.43}$$

The problem to find k_1 and k_2 is transformed into finding if there is at least one suitable $\boldsymbol{g}_3$ satisfying (5.43), so that $\boldsymbol{q}_3$ will be the linear combination of $\boldsymbol{q}_1$ and $\boldsymbol{q}_2$. Consider $\boldsymbol{\Omega}_1, \boldsymbol{\Omega}_3$ by their column vectors, it can be obtained

$$\begin{aligned} \boldsymbol{\Omega}_1 \boldsymbol{g}_1 &= [\boldsymbol{\Omega}_{11} \quad \boldsymbol{\Omega}_{12}] \begin{bmatrix} g_1[1] \\ g_1[2] \end{bmatrix} \\ &= g_1[1]\boldsymbol{\Omega}_{11} + g_1[2]\boldsymbol{\Omega}_{12} \end{aligned} \tag{5.44}$$

where $\boldsymbol{\Omega}_{11}, \boldsymbol{\Omega}_{12}$ are the first and second column vector of $\boldsymbol{\Omega}_1$. $g_1[1], g_1[2]$ are the first and second elements of $\boldsymbol{g}_1$. Then the right side of (5.43) is the arbitrary linear combination of $\boldsymbol{\Omega}_{11}$ and $\boldsymbol{\Omega}_{12}$, representing a 2-D planar space $\boldsymbol{L}_1$ supported by $\boldsymbol{\Omega}_{11}$ and $\boldsymbol{\Omega}_{12}$. Similarly, the left side of (5.43) can be considered as a vector in the 2-D space $\boldsymbol{L}_3$ constructed by $\boldsymbol{\Omega}_{31}$ and $\boldsymbol{\Omega}_{32}$. Since $\sin\theta_3 \neq \sin\theta_1, \sin\phi_3 \neq \sin\phi_1$, it can be verified $\boldsymbol{L}_1$ is not in parallel with $\boldsymbol{L}_3$. The only possibility is that both sides in (5.43) are in parallel with the intersection vector of these 2-D spaces. So it must satisfy that

$$(\boldsymbol{\Omega}_{11} \times \boldsymbol{\Omega}_{12}^T)\boldsymbol{\Omega}_3 \boldsymbol{g}_3 = 0 \tag{5.45}$$

where '$\times$' means cross product of two vectors. By solving (5.45), the polarisation of s_3 can also be obtained. Notice that $\boldsymbol{\Omega}_{11}, \boldsymbol{\Omega}_{12}, \boldsymbol{\Omega}_{31}, \boldsymbol{\Omega}_{32}$ are real-valued vectors, the intersection vector must be a real-valued vector as well. As mentioned above, $\boldsymbol{\Omega}_3 \boldsymbol{g}_3$ is in parallel with the intersection vector. As the second element of $\boldsymbol{\Omega}_3 \boldsymbol{g}_3$ is real-valued, it can be deduced that the first element must also be real-valued, which means $\eta_3 = 0°$. Then $\boldsymbol{g}_3$ can be expressed as

$$\boldsymbol{g}_3 = \begin{bmatrix} \sin\gamma_3 \\ \cos\gamma_3 \end{bmatrix} \tag{5.46}$$

Due to the orthogonality in (5.45), we have

$$(\boldsymbol{\Omega}_{11} \times \boldsymbol{\Omega}_{12}^T)\boldsymbol{\Omega}_3 = k_3 \begin{bmatrix} -\cos\gamma_3 \\ \sin\gamma_3 \end{bmatrix} \tag{5.47}$$

Using $\boldsymbol{H}$ to denote the left side of the above equation for convenience, then

$$\boldsymbol{H} = (\boldsymbol{\Omega}_{11} \times \boldsymbol{\Omega}_{12}^T)\boldsymbol{\Omega}_3 \tag{5.48}$$

The polarisation parameter γ_3 of the ambiguity source can be found at

$$\gamma_3 = \arctan(-\frac{H[2]}{H[1]}) \tag{5.49}$$

Since the DOA and polarisation parameters $(\theta_3, \phi_3, \gamma_3, \eta_3)$ can be obtained by above equations, , whose joint steering vector is also orthogonal to the noise subspace, the ambiguity problem appears. As an example, if a DST signal comes from $(30°, 80°)$, one ambiguous source can come from $(40°, 50°)$ with polarisation parameters $(19.2°, 0°)$. For each DST signal, there will be an infinite number of ambiguous SST signal directions.

For the mixed signals scenario, although the 4-D search algorithm cannot identify the DST signals, it works for SST signals. However, an obvious problem with the algorithm is the significantly high computational complexity of the 4-D peak search process. In the next subsection, we propose a two-step algorithm, which estimates the DOAs of SST and DST signals separately with a much lower complexity.

5.3.2 The proposed two-step method

As can be seen from the name, there are two steps for the proposed method. The first step is to apply a newly proposed SST estimator to obtain the DOA and polarisation of SST signals, while the second step is to apply a specifically designed DST estimator to find the DOA of DST signals.

In the first step, we only focus on the SST signals. By exploiting the orthogonality between the joint steering vector $\boldsymbol{q}_m$ and the noise subspace $\boldsymbol{U}_n$, we have

$$\boldsymbol{U}_n^H \boldsymbol{q}_m = \boldsymbol{0} \tag{5.50}$$

In order to avoid the computationally expensive 4-D search, (5.50) can be transformed to

$$\begin{aligned} \mathbf{0} &= \boldsymbol{U}_n^H[\boldsymbol{a}_m \otimes (\boldsymbol{\Omega}_m \boldsymbol{g}_m)] \\ &= \boldsymbol{U}_n^H[(\boldsymbol{a}_m \otimes \boldsymbol{\Omega}_m)\boldsymbol{g}_m] \\ &= [\boldsymbol{U}_n^H \boldsymbol{b}_m]\boldsymbol{g}_m \end{aligned} \tag{5.51}$$

where $\boldsymbol{b}_m$ is the kronecker product between $\boldsymbol{a}_m$ and $\boldsymbol{\Omega}_m$,

$$\boldsymbol{b}_m = \boldsymbol{a}_m \otimes \boldsymbol{\Omega}_m \tag{5.52}$$

For SST signals, there is only one polarisation vector $\boldsymbol{g}_m$ from a specific direction (θ_m, ϕ_m) satisfing $[\boldsymbol{U}_n^H \boldsymbol{b}_m]\boldsymbol{g}_m = \mathbf{0}$ and (5.51) indicates that the column rank of $\boldsymbol{U}_n^H \boldsymbol{b}_m$ equals to 1. Notice that $\boldsymbol{U}_n^H \boldsymbol{b}_m$ is a $(3N - M_1 - 2M_2) \times 2$ matrix. By multiplying its Hermitian transpose on the right side, the product matrix is a 2×2 matrix with rank 1, which can be denoted as

$$\mathbf{rank}\{\boldsymbol{b}_m^H \boldsymbol{U}_n \boldsymbol{U}_n^H \boldsymbol{b}_m\} = 1 \tag{5.53}$$

where the symbol $\mathbf{rank}\{\}$ is used to denote the rank of the target matrix. As the matrix is not of full rank, we have

$$\mathbf{det}\{\boldsymbol{b}_m^H \boldsymbol{U}_n \boldsymbol{U}_n^H \boldsymbol{b}_m\} = 0 \tag{5.54}$$

where $\mathbf{det}\{\}$ represents the determinant of the matrix. By taking the inverse of (5.54), a DOA estimator for SST signals is given by

$$F_1(\theta_m, \phi_m) = \frac{1}{\mathbf{det}\{\boldsymbol{b}_m^H \boldsymbol{U}_n \boldsymbol{U}_n^H \boldsymbol{b}_m\}} \tag{5.55}$$

where the peaks of the above function give the desired DOA angles of SST signals. With the DOA information obtained, the polarisation parameters can then be estimated through another 2-D search using (5.51).

However, as analysed earlier in Section 5.3.1 and further in the following, the above SST estimator does not work when source signals are DST ones.

Firstly, the kronecker product $\boldsymbol{b}_m$ can be extended to DST signals, for the m-th DST signal, we have

$$\boldsymbol{b}_m = \boldsymbol{a}_{M_1+2m-1} \otimes \boldsymbol{\Xi}_m \tag{5.56}$$

According to (5.36), for the DST signal direction

$$\begin{aligned} & \boldsymbol{U}_n^H(\boldsymbol{a}_{M_1+2m-1} \otimes \boldsymbol{\Omega}_{M_1+2m-1}) = \boldsymbol{0} \\ \Rightarrow \quad & \boldsymbol{U}_n^H(\boldsymbol{a}_{M_1+2m-1} \otimes \boldsymbol{I}_3)\boldsymbol{\Xi}_m = \boldsymbol{0} \end{aligned} \tag{5.57}$$

$\boldsymbol{\Xi}_m$ is a 3×2 matrix and can be divided into two column vectors as

$$\boldsymbol{\Xi}_m = [\boldsymbol{\Xi}_{m1} \quad \boldsymbol{\Xi}_{m2}] \tag{5.58}$$

(5.57) indicates that all the row vectors in $\boldsymbol{U}_n^H(\boldsymbol{a}_{M_1+2m-1} \otimes \boldsymbol{I}_3)$ are orthogonal to the two column vectors $\boldsymbol{\Xi}_{m1}$ and $\boldsymbol{\Xi}_{m2}$ simultaneously. As $\boldsymbol{\Xi}_{m1}$ and $\boldsymbol{\Xi}_{m2}$ are 3×1 vectors and they are not in parallel, all the row vectors in $\boldsymbol{U}_n^H(\boldsymbol{a}_{M_1+2m-1} \otimes \boldsymbol{I}_3)$ should be parallel with each other, which means the row rank of $\boldsymbol{U}_n^H(\boldsymbol{a}_{M_1+2m-1} \otimes \boldsymbol{I}_3)$ equals 1. Given the pair of angles (θ_e, ϕ_e) where there is no actually signal coming but it satisfies $\sin\theta_e \sin\phi_e = \sin\theta_m \sin\phi_m$, it can be obtained that the direction has the same angular steering vector as the DST signal, i.e. $\boldsymbol{a}_e = \boldsymbol{a}_{M_1+2m-1}$. It can be further deduced that

$$\mathbf{det}\{\boldsymbol{b}_e^H \boldsymbol{U}_n \boldsymbol{U}_n^H \boldsymbol{b}_e\} = 0 \tag{5.59}$$

As a result, this new direction will also be recognised as a peak in the spectrum even if there is no signal coming from that direction. It's easy to verify that there are infinite number of directions satisfying $\sin\theta_e \sin\phi_e = \sin\theta_n \sin\phi_n$ and these directions provide ambiguity in the estimation results. To solve the ambiguity problem of the SST estimator, in the following we propose a new estimator to deal with the DST signals, which is the second step of the proposed two-step method.

As mentioned, a DST signal $\boldsymbol{s}_m$ includes two sub-signals s_{M_1+2m-1} and s_{M_1+2m} with different polarisations, which means $\boldsymbol{g}_{M_1+2m-1}$ and $\boldsymbol{g}_{M_1+2m}$ are not in parallel with each other. Since $\boldsymbol{g}_{M_1+2m-1}$ and $\boldsymbol{g}_{M_1+2m}$ are both 2×1 column vectors and each row vector in

$\boldsymbol{U}_n^H \boldsymbol{b}_{M_1+2m-1}$ is of 1×2 orthogonal to $\boldsymbol{g}_{M_1+2m-1}$ and $\boldsymbol{g}_{M_1+2m}$, it can be deduced that all the row vectors must be zero-vectors and $\boldsymbol{U}_n^H \boldsymbol{b}_m$ is a zero-matrix as indicated by (5.57). Comparing to SST signals, the rank of $\boldsymbol{b}_m \boldsymbol{U}_n \boldsymbol{U}_n^H \boldsymbol{b}_m$ for DST signals equals to 0 instead of 1. Hence, the following cost function can be used to estimate directions of DST signals

$$F_2(\theta_n, \phi_n) = \frac{1}{||\boldsymbol{b}_m^H \boldsymbol{U}_n \boldsymbol{U}_n^H \boldsymbol{b}_m||_2} \tag{5.60}$$

where $||\boldsymbol{x}||_2$ denotes the l_2-norm of vector $\boldsymbol{x}$.

This estimator solves the DOA estimation problem of DST signals. However, when the DST estimator is applied to a mixture of SST and DST signals, the estimator (5.60) only selects the directions with

$$\mathbf{rank}\{\boldsymbol{b}_m^H \boldsymbol{U}_n \boldsymbol{U}_n^H \boldsymbol{b}_m\} = 0 \tag{5.61}$$

However, for SST signals, (5.53) indicates that the rank of $\boldsymbol{b}_m^H \boldsymbol{U}_n \boldsymbol{U}_n^H \boldsymbol{b}_m$ equals 1, which results in the loss of SST directions in the DST estimator.

The following is a summary to the proposed two-step method:

- Calculate the noise subspace $\boldsymbol{U}_n$ by applying eigenvalue decomposition to the estimated covariance matrix $\hat{\boldsymbol{R}}$.
- Apply the SST estimator (5.55). Based on the DOA results obtained from the SST estimator, find the DOAs of SST signals by 2-D search.
- Find the polarisation parameters of SST signals using (5.51) by 2-D search.
- Apply (5.60) to estimate the DOAs of DST signals.

5.3.3 The proposed general MST estimator

Instead of employing separate estimators for SST and DST signals, in this section we propose a single general estimator for MST signals, which has an better performance in terms of estimation accuracy.

Before introducing the general estimator, we first investigate the rank and determinant of the two matrices $(\boldsymbol{a}_m \otimes \boldsymbol{I}_3)^H \boldsymbol{U}_n \boldsymbol{U}_n^H (\boldsymbol{a}_m \otimes \boldsymbol{I}_3)$ and $\boldsymbol{b}_m^H \boldsymbol{U}_n \boldsymbol{U}_n^H \boldsymbol{b}_m$. In the following, for convenience we drop the subscript m and denote the two matrices as

$$\begin{aligned} \boldsymbol{A} &= (\boldsymbol{a} \otimes \boldsymbol{I}_3)^H \boldsymbol{U}_n \boldsymbol{U}_n^H (\boldsymbol{a} \otimes \boldsymbol{I}_3) \\ \boldsymbol{B} &= \boldsymbol{b}^H \boldsymbol{U}_n \boldsymbol{U}_n^H \boldsymbol{b} \end{aligned} \tag{5.62}$$

In the scenario with a mixture of SST and DST signals, we can divide the direction range into four regions: the SST signal direction region, the DST signal direction region, the DST ambiguity direction region and the remaining uninterested direction region. Table 5.1 gives a summary of the ranks of A and B and the their associated direction regions.

Table 5.1: rank A and B for different direction regions

	Rank(A)	Rank(B)
SST Signal	2	1
DST Signal	1	0
DST Ambiguity	1	1
Uninterested	3	2

As discussed before, the SST estimator selects the direction with the condition $\mathbf{rank}\{\boldsymbol{B}\} < 2$ and the DST estimator selects the direction with $\mathbf{rank}\{\boldsymbol{B}\} = 0$. From Table 5.1, we can see that the SST estimator can find the SST signal directions, DST signal directions and the DST ambiguity directions while the DST estimator only estimates the DST signal directions. As a solution to the problem, we propose a general MST estimator which can work in all cases of signals and its cost function is given by

$$F_3(\theta, \phi) = \frac{\sum_{i=1}^{3} \sum_{j=1}^{3} \mathbf{det}\{\boldsymbol{A}_{i,j}\}}{\mathbf{det}\{\boldsymbol{B}\}} \tag{5.63}$$

where $\boldsymbol{A}_{i,j}$ is the cofactor matrix of matrix $\boldsymbol{A}$ by removing its $i-th$ row and $j-th$ column. The estimator is able to estimate DOA information for all signals without determining its

type, i.e. SST or DST. After obtaining all the DOAs, we can then use (5.51) to find the polarisation parameters of SST signals through 2-D search if needed. For DST signals, we can distinguish them from the SST ones by checking whether there is polarisation ambiguity problem or not when doing the search using (5.51).

The reason why the above cost function works can be explained roughly as follows. First, note that matrix $\boldsymbol{A}$ is a 3×3 matrix and its cofactor matrix is of 2×2. For the SST signal direction region, the rank of matrix $\boldsymbol{A}$ is 2 and then one of its cofactor matrices must have a rank of 2 so that its determinant is nonzero and the summation in the numerator is nonzero, while matrix $\boldsymbol{B}$ has a rank of 1 and its determinant is zero; as a result, the cost function at the directions of SST signals will have a peak (an infinitely large value in theory). For the DST direction region, the rank of matrix $\boldsymbol{A}$ is 1 and then one of its cofactor matrices must have a rank of 1 and non-zero-valued, so that although its determinant is zero and the summation in the numerator is also zero, it approaches zero at those directions at the first order, while the 2×2 matrix $\boldsymbol{B}$ has a rank of 0 and its determinant is zero and approaches zero at those directions at the second order (a 2×2 zero matrix); as a result, the cost function at the directions of DST signals will have a peak too (an infinitely large value in theory). For the DST ambiguity region, the rank of matrix $\boldsymbol{A}$ is 1 and similar to the case of DST direction region, the summation in the numerator of (5.63) is zero, but it approaches zero at those ambiguity directions at the first order, while the 2×2 matrix $\boldsymbol{B}$ has a rank of 1 and its determinant is zero and approaches zero at those directions at the first order (a 2×2 nonzero matrix); as a result, the cost function at the DST ambiguity region will be a nonzero finite value, but not a peak representing an infinitely large value. For the uninterested region, both matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ have full rank and neither of the numerator and denominator of the cost function is zero-valued; as a result, the cost function at the uninterested direction region will be a nonzero finite value, but not a peak representing an infinitely large value.

A detailed proof can be found in the Appendix.

A summary for the unified general MST estimator is give below:

- Calculate the noise space $\boldsymbol{U}_n$ by applying eigenvalue decomposition to the estimated covariance matrix $\hat{\boldsymbol{R}}$.
- Apply the MST estimator (5.55) to obtain the DOA of all signals by 2-D search.
- Find the polarisation parameters of SST signals by (5.51).

5.4 Cramér-Rao Bound for MST Signals

In this section, we derive the CRB for DOA estimation of a mixture of one SST signal and one DST signal to evaluate the performance of the proposed algorithms. A basic assumption is that all source signals are unconditional [85], which means the source signals are random in all realizations. The SST signal and the two DST sub-signals have the same power σ_s^2.

Here we use the symbol $\boldsymbol{\alpha}$ to denote the parameters to be estimated, which can be denoted as

$$\boldsymbol{\alpha} = (\theta_1, \theta_2, \phi_1, \phi_2) \tag{5.64}$$

where (θ_1, ϕ_1) is the DOA parameters for the SST signal and (θ_2, ϕ_2) is the parameters for the DST signal. Note that in the proposed estimators, the DOA parameters are obtained separately from the polarisation parameters. This means in the DOA estimation process, the polarisation parameters can be considered as kind of irrelevant parameters. From (5.16), the received signals can be changed to

$$\begin{aligned} \boldsymbol{y}[k] &= \sum_{m=1}^{M_1+2M_2} \boldsymbol{a}_m \otimes \boldsymbol{\Omega}_{\mathbf{m}} \boldsymbol{g}_m \cdot s_m[k] + \boldsymbol{n}[k] \\ &= (\sum_{m=1}^{M_1+2M_2} \boldsymbol{b}_m \cdot s_m[k] + \boldsymbol{n}[k] \boldsymbol{g}_m^H) \boldsymbol{g}_m \end{aligned} \tag{5.65}$$

The equation holds because

$$\boldsymbol{g}_m^H \boldsymbol{g}_m = 1 \tag{5.66}$$

Define a matrix of the received signals $\boldsymbol{Z}[k]$, where

$$\boldsymbol{Z}[k] = \sum_{m=1}^{M_1+2M_2} \boldsymbol{b}_m \cdot s_m[k] + \boldsymbol{n}[k]\boldsymbol{g}_m^H \tag{5.67}$$

In the proposed two methods, we estimate DOA parameters (θ, ϕ) by $\boldsymbol{Z}[k]$ instead of $\boldsymbol{y}[k]$. For each snapshot, the probability density function is given by [73]

$$p_z|(\boldsymbol{\alpha}) = \frac{1}{\det[\pi\, \boldsymbol{V_Z}(\boldsymbol{\alpha})]} e^{\{-[\boldsymbol{Z}-\boldsymbol{m}(\boldsymbol{\alpha})]^H\, \boldsymbol{V}_Z^{-1}(\boldsymbol{\alpha})[\boldsymbol{Z}-\boldsymbol{m}(\boldsymbol{\alpha})]\}} \tag{5.68}$$

where $\boldsymbol{V_Z}(\boldsymbol{\alpha})$ is variance of $\boldsymbol{Z}$ and $\boldsymbol{m}(\boldsymbol{\alpha})$ is the mean value.

The joint probability density function with K snapshots can be denoted as

$$\begin{aligned} p_{Z_1,Z_2,\ldots,Z_K}|(\boldsymbol{\alpha}) = & \prod_{k=1}^{K} \frac{1}{\det[\pi\, \boldsymbol{V_Z}(\boldsymbol{\alpha})]} \\ & \cdot e^{\{-[\boldsymbol{Z}_k-\boldsymbol{m}(\alpha)]^H\, \boldsymbol{V}_Z^{-1}(\alpha)[\boldsymbol{Z}_k-\boldsymbol{m}(\boldsymbol{\alpha})]\}} \end{aligned} \tag{5.69}$$

which leads to the following log-likelihood function

$$\begin{aligned} L_x(\boldsymbol{\alpha}) = & \ln p_{Z_1,Z_2,\ldots,Z_K}|(\boldsymbol{\alpha}) \\ = & -K \ln \det[\,\boldsymbol{V_Z}(\boldsymbol{\alpha})] - KN \ln \pi \\ & - \sum_{k=1}^{K} [\boldsymbol{Z}_k - \boldsymbol{m}(\boldsymbol{\alpha})]^H\, \boldsymbol{V}_Z^{-1}(\boldsymbol{\alpha})[\boldsymbol{Z}_k - \boldsymbol{m}(\boldsymbol{\alpha})] \end{aligned} \tag{5.70}$$

The elements in the fisher information matrix (FIM) can be found as

$$\begin{aligned} F_{\alpha_i,\alpha_j} = & E[\frac{\partial L_Z(\boldsymbol{\alpha})}{\partial \alpha_i} \cdot \frac{\partial L_Z(\boldsymbol{\alpha})}{\partial \alpha_j}] \\ = & -E[\frac{\partial^2 L_Z(\boldsymbol{\alpha})}{\partial \alpha_i \partial \alpha_j}] \end{aligned} \tag{5.71}$$

where i, j are integers and $i, j \in [1, 8]$.

According to (8.32) in [73], (5.71) can be simplified to

$$\begin{aligned} F_{\alpha_i,\alpha_j} = & \mathrm{tr}\{\boldsymbol{R}_{\boldsymbol{Z}}^{-1}(\boldsymbol{\alpha}) \frac{\partial \boldsymbol{R_Z}(\boldsymbol{\alpha})}{\alpha_i} \boldsymbol{R}_{\boldsymbol{Z}}^{-1}(\boldsymbol{\alpha}) \frac{\partial \boldsymbol{R_Z}(\boldsymbol{\alpha})}{\alpha_j}\} \\ & + 2Re\{\frac{\partial \boldsymbol{m}^H(\boldsymbol{\alpha})}{\alpha_i} \boldsymbol{R}_{\boldsymbol{Z}}^{-1}(\boldsymbol{\alpha}) \frac{\partial \boldsymbol{m}(\boldsymbol{\alpha})}{\alpha_j}\} \end{aligned} \tag{5.72}$$

Since the source signals are unconditional, we have

$$\boldsymbol{m}(\boldsymbol{\alpha}) = \boldsymbol{0} \tag{5.73}$$

and

$$\begin{aligned}\boldsymbol{V_Z} =&\sigma_s^2 \boldsymbol{b}_1 \boldsymbol{b}_1^H + \sigma_s^2 \boldsymbol{b}_2 \boldsymbol{b}_2^H + \sigma_s^2 \boldsymbol{b}_3 \boldsymbol{b}_3^H + \sigma_n^2 \boldsymbol{I} \\ =&\sigma_s^2 \boldsymbol{b}_1 \boldsymbol{b}_1^H + 2\sigma_s^2 \boldsymbol{b}_2 \boldsymbol{b}_2^H + \sigma_n^2 \boldsymbol{I}\end{aligned} \tag{5.74}$$

The FIM elements are transformed to

$$F_{\alpha_i,\alpha_j} = \mathrm{tr}\{ \boldsymbol{V_Z}^{-1}(\boldsymbol{\alpha}) \frac{\partial \boldsymbol{V_Z}(\boldsymbol{\alpha})}{\alpha_i} \boldsymbol{V_Z}^{-1}(\boldsymbol{\alpha}) \frac{\partial \boldsymbol{V_Z}(\boldsymbol{\alpha})}{\alpha_j} \} \tag{5.75}$$

The fisher information matrix is a 4×4 matrix. The CRB for DOA information can be obtained as

$$\begin{aligned} CRB(\theta_1) &= [\boldsymbol{F}^{-1}(\boldsymbol{\alpha})]_{1,1} \\ CRB(\theta_2) &= [\boldsymbol{F}^{-1}(\boldsymbol{\alpha})]_{2,2} \\ CRB(\phi_1) &= [\boldsymbol{F}^{-1}(\boldsymbol{\alpha})]_{3,3} \\ CRB(\phi_2) &= [\boldsymbol{F}^{-1}(\boldsymbol{\alpha})]_{4,4} \end{aligned} \tag{5.76}$$

The CRB for only SST signals and only DST signals can be obtained by simply removing the DST or SST signal part in (5.67), and the fisher information matrix will be reduced to a 2×2 matrix.

5.5 Simulation Results

In this section, simulations are performed based on a scenario with one SST signal and one DST signal impinging on the array from the far field.

5.5.1 DOA spectrum

Consider a uniform linear tripole sensor array with $M = 5$ sensors and the inter-element distance d is set to half wavelength $\lambda/2$. The SST signal and each sub-signal of a

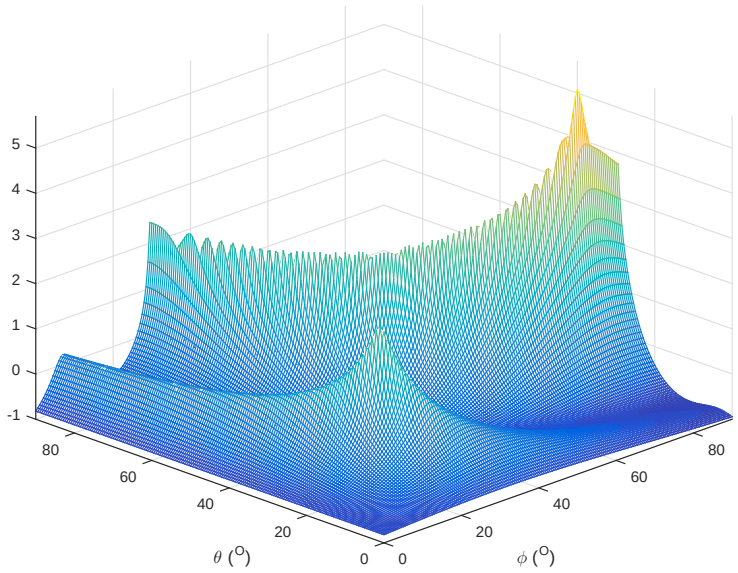

Figure 5.2: DOA spectrum of SST estimator.

DST signal have the same power σ_s^2 with $SNR = 10$ dB. The SST signal comes from $(\theta_1, \phi_1, \gamma_1, \eta_1) = (20°, 20°, 50°, 10°)$ and DST signal comes from $(\theta_2, \phi_2, \gamma_2, \eta_2, \gamma_3, \eta_3) = (30°, 80°, 20°, 50°, 70°, -40°)$. The total number of snapshots is set to 1000 and the searching stepsize is set to 0.5°. The spatial spectrum results obtained by applying our proposed two-step method and the one-step general method are shown in Fig. 5.2, Fig. 5.3 and Fig. 5.4.

Fig. 5.2 and Fig. 5.3 are for the results by two-step method. IThe SST estimator result is shown in Fig. 5.2, where the peak corresponding to the SST signal appears around the aimed direction $(\theta, \phi) = (20°, 20°)$; however, the DST signal direction is shown among a band of peak points instead of single peak as discussed in the former section. On the other hand, the second step focuses on locating DST signals and as shown in Fig.5.3, only peak appears around the aimed DST signal direction $(\theta, \phi) = (30°, 80°)$ while the SST signal direction is lost in the spectrum.

For the one-step general estimator or the so-called MST estimator, the spectrum have two peaks at around $(\theta, \phi) = (20°, 20°)$ and $(30°, 80°)$, indicating the directions of the SST signal and DST signals have been identified successfully.

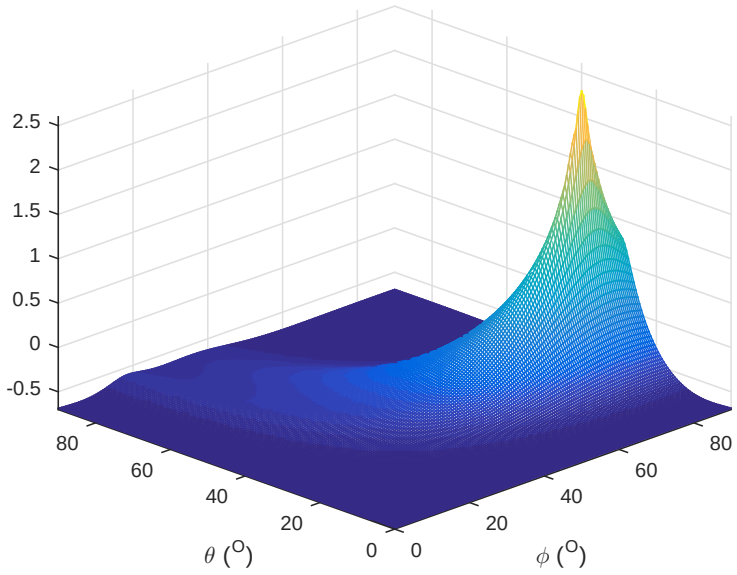

Figure 5.3: DOA spectrum of DST estimator.

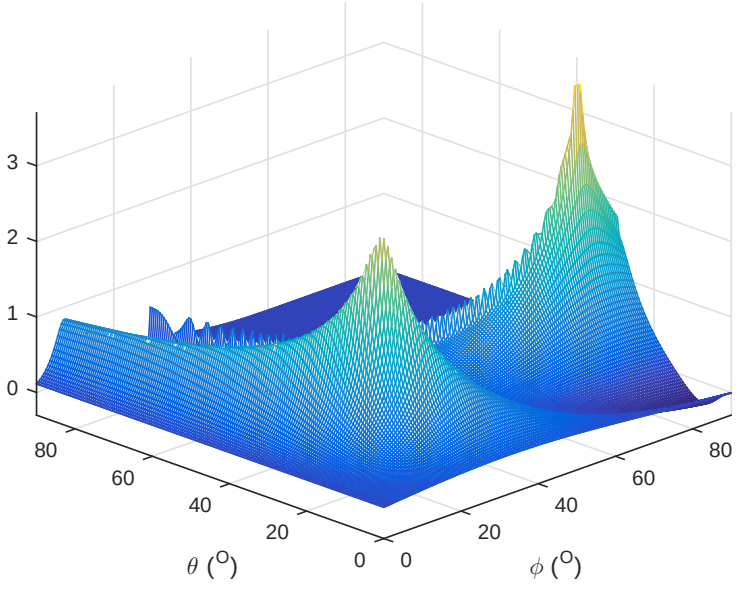

Figure 5.4: DOA spectrum of MST estimator.

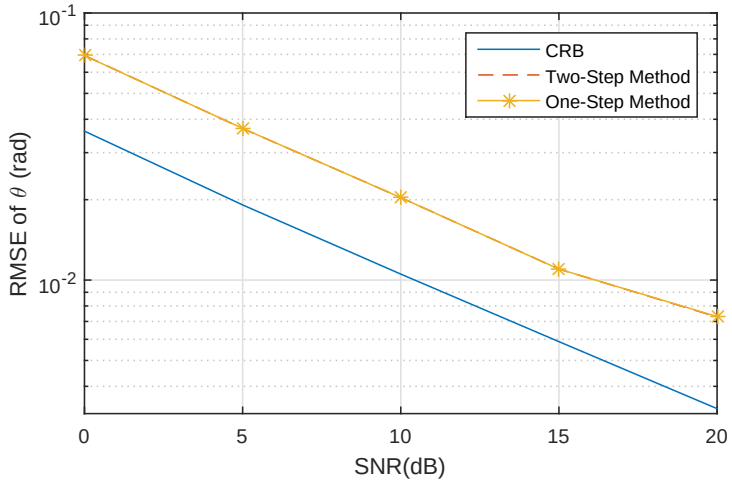

Figure 5.5: RMSE of elevation angle θ versus SNR, SST signal only.

5.5.2 RMSE result

In this part, we compare the estimation accuracy for the two proposed solutions. Three scenarios are considered: only one SST signal is present; only one DST signal is present; a mixture of one SST signal and one DST signal. In these scenarios, the directions of SST and DST signals are the same as in Section V-A, and the power of SST signal is equal to that of one DST sub-signal. We calculate the RMSE (root mean square error) of the azimuth-elevation angle θ, ϕ by 200 Monte-Carlo trials. The number of snapshots is $K = 100$ and the searching step size $0.05°$. The RMSE of θ, ϕ is defined as follows,

$$RMSE_{\theta_m} = \sqrt{\frac{1}{200}\sum_{t=1}^{200}(\theta_t - \theta_m)^2} \tag{5.77}$$

and

$$RMSE_{\phi_m} = \sqrt{\frac{1}{200}\sum_{t=1}^{200}(\phi_t - \phi_m)^2} \tag{5.78}$$

where ϕ_t is the estimated value in the $t-th$ trial and θ_m, ϕ_m is the actual value.

In the first scenario, only one SST signal impinges on the array. As shown in Fig. 5.5 and Fig. 5.6, the two methods have almost the same estimating accuracy and the RMSE of both estimator decrease gradually with the increasing SNR.

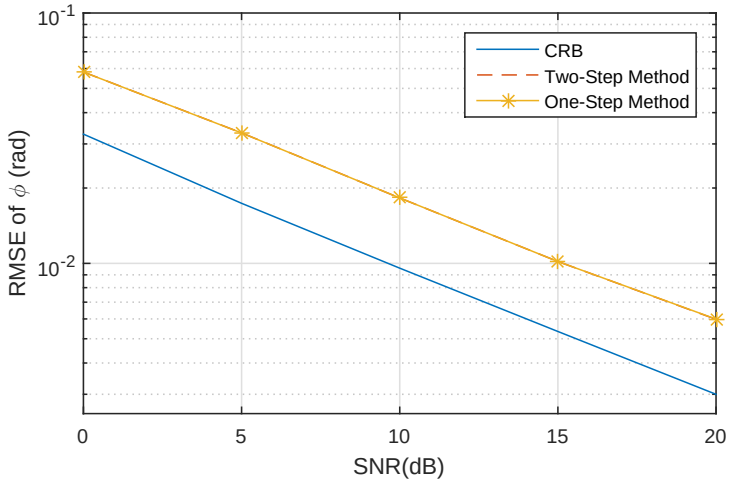

Figure 5.6: RMSE of azimuth angle ϕ versus SNR, SST signal only.

In the second scenario with DST signals, the results are presented in Fig. 5.7 and Fig. 5.8. Compared to the SST signal case, the DST case has lower average estimation errors, and the general one-step method has a higher accuracy than the two-step method.

In the last scenario, it has one SST signal s_1 from $(20°, 20°)$ and one DST signal $\boldsymbol{s}_2$ from $(30°, 80°)$, and the RMSE results are shown in Fig. 5.9 and Fig. 5.10. Compared to Fig. 5.5 and Fig. 5.6, the estimation error increases a little due to the additional DST signal. Fig. 5.11 and Fig. 5.12 also indicates the same difference versus Fig. 5.7 and Fig. 5.8. In this scenario, the two proposed methods still have a very similar performance in estimating the SST signal direction. However, the general one-step method has lower RMSE than the two-step method with the DST signals.

5.6 Summary

In this Chapter, the DOA estimation problem for a mixture of SST and DST signals has been studied based on a tripole linear array. Two subspace based DOA estimation methods were proposed and the CRB was derived to evaluate their performance. The two-step method estimates the SST and DST signals' directions separately with two corresponding estimators, one for the SST signals and one for the DST signals. The

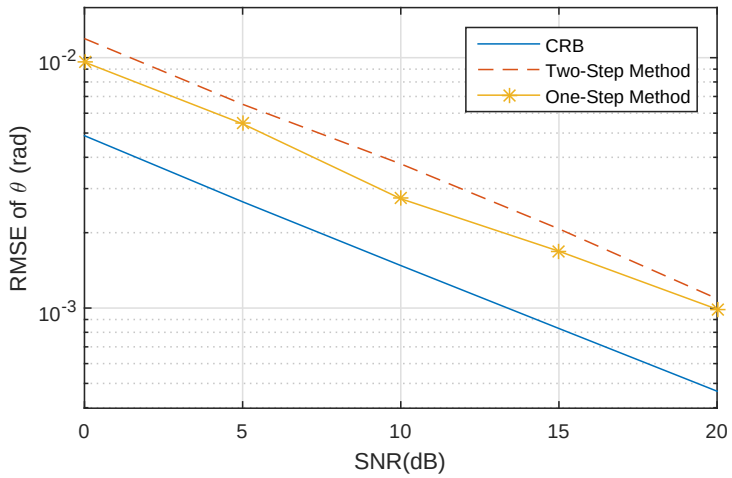

Figure 5.7: RMSE of elevation angle θ versus SNR, DST signal only.

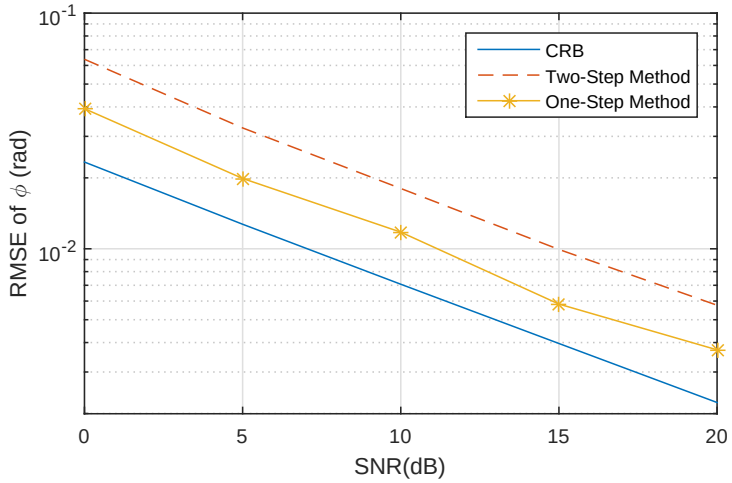

Figure 5.8: RMSE of azimuth angle ϕ versus SNR, DST signal only.

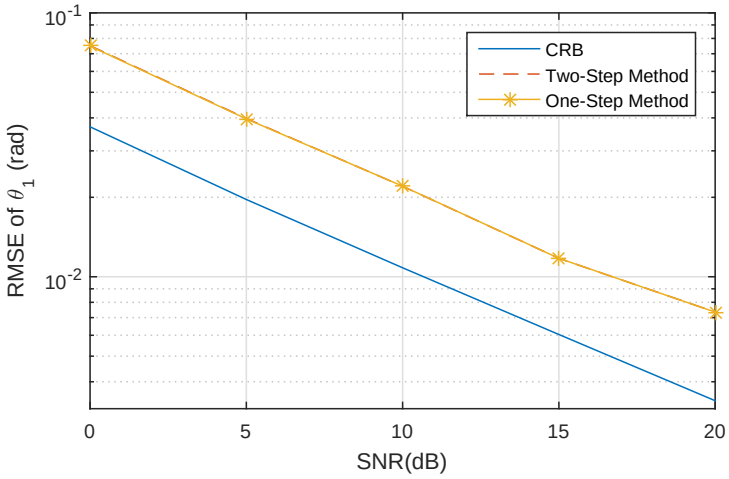

Figure 5.9: RMSE for SST signal elevation angle θ_1 versus SNR, mixed signals.

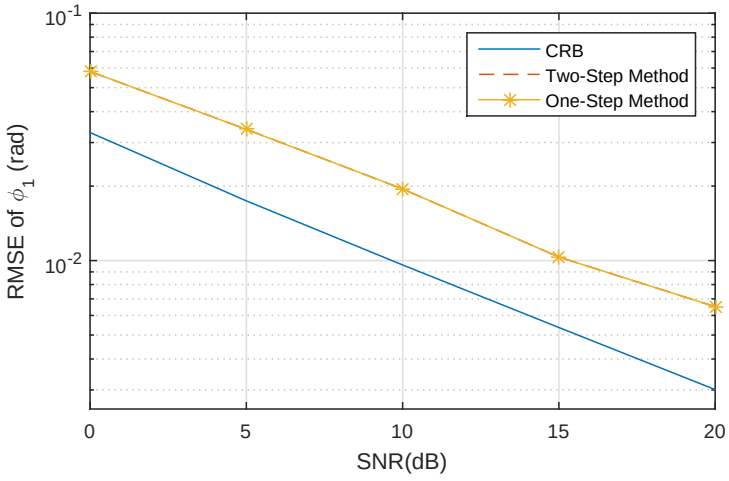

Figure 5.10: RMSE of SST azimuth angle ϕ_1 versus SNR, mixed signals.

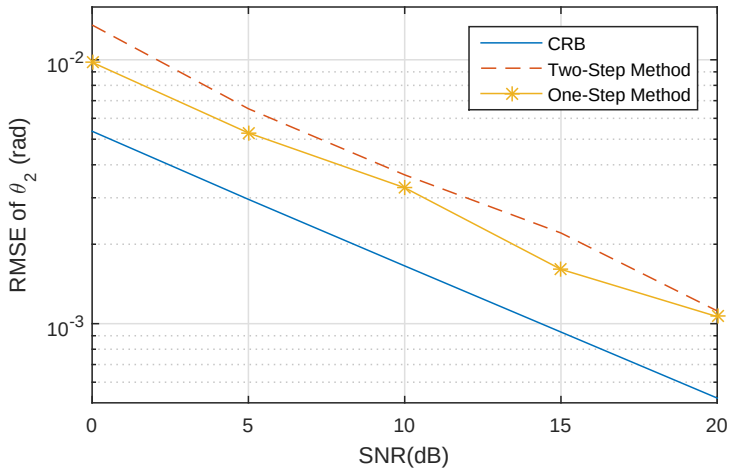

Figure 5.11: RMSE of DST elevation angle θ_2 versus SNR, mixed signals.

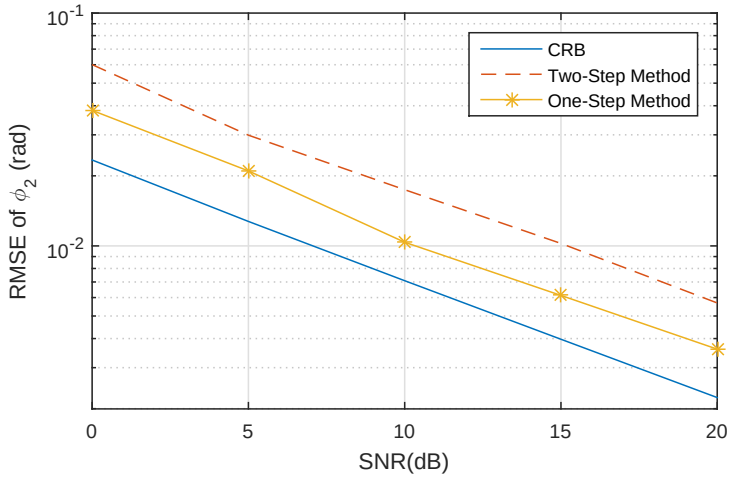

Figure 5.12: RMSE of DST azimuth angle ϕ_2 versus SNR, mixed signals.

second method is a general one-step method which estimates the signal directions together without distinguishing the different types of signals. Simulation results showed that the proposed two methods have a very similar performance for SST signals, but the one-step method has some advantages in dealing with DST signals.

Chapter 6

Conclusions and Future Plan

6.1 Conclusions

In this thesis, we first gave a review of some basic concepts and related techniques in array signal processing in Chapter 2 and then focused on three main contributions in the following three chapters, respectively.

In Chapter 3, a fully quaternion-valued adaptive beamformer was first proposed. The existing quaternion-capon beamformer only changes the steering vector into the quaternion field with the polarisation parameters. A problem is that the signal itself is still complex-valued so that the beamformer is not a full quaternion model in maths, leading to unnecessarily high computational complexity. To solve the problem, two separate signals s_1 and s_2 from the same direction but different polarisations are considered as a quaternion signal $q = s_1 + is_2$. Based on this, a full quaternion capon beamformer is proposed, with the received signals constrained to the form of $s_1 + is_2$. The main contribution is to make full use of the storage, avoiding wasting the i, k parts as in quaternion beamformer. Analysis on computation complexity shows the full quaternion beamformer has lower complexity than the existing quaternion beamformer. Moreover, the full quaternion Capon beamformer was further developed into a robust one to tackle various array model errors.

As studied in Chapter 4, for the joint DOA and polarisation estimation problem, the ambiguity phenomenon cannot be avoided when linear crossed-dipole sensor array is employed and a proof was given to explain why this ambiguity occurs in this case. To solve the problem, a tripole senor array can be used as supported by detailed analysis. With tripole sensor array, the joint estimation problem can be solved by a 4-D MUSIC algorithm. Unfortunately, the 4-D peak search has an extremely high computational complexity. In order to reduce the complexity and avoid 4-D search, the complexity reduction MUSIC algorithm was proposed, where the 4-D search is replaced by two 2-D searches. However, simulation shows that complexity reduction is achieved at the cost of estimation accuracy, and the 4-D search algorithm has a higher accuracy.

In Chapter 5, the DOA estimation problem for a mixture of SST and DST signals was studied for the first time. An analysis was first given to show that for SST signals, both DOA and polarisation parameters can be estimated. However, for DST signals, only their DOA can be estimated while the polarisation parameters cannot due to inherent property of the DST signal model. To estimate the DOAs of both SST and DST signals, two solutions were proposed. One is a two-step method to separately estimate the DOAs of SST and DST signals. The other is a unified one-step method to estimate the DOAs of all signals together. Simulations were provided to compare the performance between the two solutions versus CRB. As shown, the two methods had a similar performance for SST signals while for DST signals, the one-step method outperformed the two-step method.

6.2 Future Work

In our past research, most estimation work is based on the long-vector maths model. As introduced in literature review and the quaternion beamformer chapter, the received signal model can also be constructed by a quaternion model. In the future, the quaternion model can be applied to the DOA estimation work. The difference between these two models can be analyzed and it will be a nice work to find out the reason which leads to the difference.

Another idea for the future work is to investigate a rotating sensor array. For example, in a tripole sensor array, each sensor rotates with a constant angle compared to the last sensor. When applied for DOA estimation, by analysing the CRB between the new sensor array and the traditional one, it can be obtained which structure has a higher estimation accuracy.

After that, another interesting topic is the DOA estimation and the related analysis of dual-transmission signals. We have proposed a MUSIC like algorithm in the paper to solve the problem, however, our algorithm is based on an assumption that the two sub-signals are uncorrelated. If the signals are correlated, the MUSIC like algorithm may fail to estimate the DOA. To solve the problem, it's reasonable to consider applying other algorithms, for example, the compressive sensing (CS) algorithm for correlated DOA estimation. However, since our estimation is based on polarised signals, the DOA and polarisation parameters appear in 4-D vectors. The existing CS algorithm only considers 1-D direction parameter. If 4-D parameters are applied in the algorithm, it's very important to find a way to reduce size of the searching matrix.

The DOA estimation problem for DST signals is still not widely studied. There may exist many problems when apply DST signals directly in current communication system. Our current research has found out the ambiguity problem due to the same DOA of the two sub-signals. It's believed that there are still many unknown problems in this area and it's worth to taking efforts to study in this field. I hope one day, DST signals could be widely used in our communication system, which may be a great revolution for our modern life.

Appendix

Proof of the MST Estimator

The appendix is to explain how MST estimator works in sec 5.3.3. For convenience, in the next we use x to denote $\sin\theta$ and y to denote $\sin\phi$. Then, the steering vector $\boldsymbol{a}$ can be expressed by

$$\begin{aligned} \boldsymbol{a} &= [1, e^{-j\tau xy}, ..., e^{-j(N-1)\tau xy}]^T \\ &= [1, C^{xy}, C^{2xy}, ..., C^{(N-1)xy}] \end{aligned} \tag{1}$$

where $C = e^{-j\tau}$ is a constant. The angular matrix $\boldsymbol{\Omega}$ becomes

$$\boldsymbol{\Omega} = \begin{bmatrix} \sqrt{1-x^2}\cdot\sqrt{1-y^2} & -y \\ \sqrt{1-x^2}\cdot y & \sqrt{1-y^2} \\ -x & 0 \end{bmatrix} \tag{2}$$

Adding an infinitely small value $\Delta x \to 0$ and $\Delta y \to 0$ to x and y, respectively, the new steering vector $\hat{\boldsymbol{a}}$ becomes

$$\begin{aligned} \hat{\boldsymbol{a}} &= [1, C^{2(xy+x\Delta y+y\Delta x+\Delta x\Delta y)}, \\ &\quad ..., C^{(N-1)(xy+x\Delta y+y\Delta x+\Delta x\Delta y)}]^T \end{aligned} \tag{3}$$

We use $\bar{\boldsymbol{a}}$ to denote the difference between the two vectors

$$\bar{\boldsymbol{a}} = \hat{\boldsymbol{a}} - \boldsymbol{a} \tag{4}$$

Its n-th element $\bar{a}_n, n \in [1, N]$ is expressed as

$$\bar{a}_n = C^{(n-1)xy}[C^{(n-1)(x\Delta y+y\Delta x+\Delta x\Delta y)} - 1] \tag{5}$$

Similarly, the difference between the original and the new angular matrix $\bar{\mathbf{\Omega}}$ can be also calculated by

$$\bar{\mathbf{\Omega}} = \hat{\mathbf{\Omega}} - \mathbf{\Omega} \tag{6}$$

where

$$\begin{aligned}
\bar{\Omega}_{11} &= \sqrt{1-(x+\Delta x)^2}\cdot\sqrt{1-(y+\Delta y)^2} \\
&- \sqrt{1-x^2}\cdot\sqrt{1-y^2} \\
\bar{\Omega}_{12} &= -\Delta y \\
\bar{\Omega}_{21} &= \sqrt{1-(x+\Delta x)^2}\cdot(y+\Delta y) - \sqrt{1-x^2}\cdot y \\
\bar{\Omega}_{22} &= \sqrt{1-(y+\Delta y)^2} - \sqrt{1-y^2} \\
\bar{\Omega}_{31} &= -\Delta x \\
\bar{\Omega}_{31} &= 0
\end{aligned} \tag{7}$$

The differences $\bar{\boldsymbol{a}}$ and $\bar{\mathbf{\Omega}}$ lead to changes of matrices $\boldsymbol{A}$ and $\boldsymbol{B}$. The changed matrices are

$$\begin{aligned}
\hat{\boldsymbol{A}} &= [(\boldsymbol{a}+\bar{\boldsymbol{a}})\otimes\boldsymbol{I}_3]^H\,\boldsymbol{U}_n\,\boldsymbol{U}_n^H[(\boldsymbol{a}+\bar{\boldsymbol{a}})\otimes\boldsymbol{I}_3] \\
\hat{\boldsymbol{B}} &= \hat{\mathbf{\Omega}}^H\hat{\boldsymbol{A}}\hat{\mathbf{\Omega}} \\
&= (\mathbf{\Omega}+\bar{\mathbf{\Omega}})^H\hat{\boldsymbol{A}}(\mathbf{\Omega}+\bar{\mathbf{\Omega}})
\end{aligned} \tag{8}$$

Replacing $\boldsymbol{A}$, $\boldsymbol{B}$ by $\hat{\boldsymbol{A}}$, $\hat{\boldsymbol{B}}$ in (5.63), we have

$$\begin{aligned}
F_3(x+\Delta x, y+\Delta y) &= \frac{\sum_{i=1}^{3}\sum_{j=1}^{3}\mathbf{det}\{\hat{\boldsymbol{A}}_{i,j}\}}{\mathbf{det}\{\hat{\boldsymbol{B}}\}} \\
&= \frac{\sum_{i=1}^{3}\sum_{j=1}^{3}(\mathbf{det}\{\boldsymbol{A}_{i,j}\}+v_{i,j})}{\mathbf{det}\{\boldsymbol{B}\}+w}
\end{aligned} \tag{9}$$

where $v_{i,j}$ and w are the determinant differences between the original and the changed matrices. When $\Delta x, \Delta y \to 0$, $v_{i,j}$ and w also approach 0.

Now we consider the four cases listed in Table 5.1.

Case 1: For SST signal directions, $\mathbf{rank}\{A\} = 2$ and $\mathbf{rank}\{B\} = 1$. The sum of the determinant for each cofactor $\sum_{i=1}^{3}\sum_{j=1}^{3}\mathbf{det}\{\boldsymbol{A}_{i,j}\}$ must be non-zero. As $v_{i,j} \to 0$, the numerator of the estimator approaches a non-zero constant. Since $\mathbf{rank}\{B\} = 1$, we have $\mathbf{det}\{B\} = 0$. As $w \to 0$, the denominator of the estimator approaches 0. Hence for SST signal directions, the estimator will have an infinitely large value and the directions will be detected by the estimator as peaks.

Case 2: Expanding equation (8), we have

$$\begin{aligned}\hat{\boldsymbol{A}} = &\ (\boldsymbol{a}\otimes\boldsymbol{I}_3)^H\boldsymbol{T}_n(\boldsymbol{a}\otimes\boldsymbol{I}_3) + (\bar{\boldsymbol{a}}\otimes\boldsymbol{I}_3)^H\boldsymbol{T}_n(\boldsymbol{a}\otimes\boldsymbol{I}_3)\\ + &\ (\boldsymbol{a}\otimes\boldsymbol{I}_3)^H\boldsymbol{T}_n(\bar{\boldsymbol{a}}\otimes\boldsymbol{I}_3) + (\bar{\boldsymbol{a}}\otimes\boldsymbol{I}_3)^H\boldsymbol{T}_n(\bar{\boldsymbol{a}}\otimes\boldsymbol{I}_3)\end{aligned} \tag{10}$$

where

$$\boldsymbol{T}_n = \boldsymbol{U}_n\boldsymbol{U}_n^H \tag{11}$$

For DST signals, the direction parameters are already known. In (10), $\boldsymbol{a}$ and $\boldsymbol{T}$ are vector and matrix with constant value elements. We have the difference matrix $\bar{\boldsymbol{A}}$ as

$$\begin{aligned}\bar{\boldsymbol{A}} = &\ (\bar{\boldsymbol{a}}\otimes\boldsymbol{I}_3)^H\boldsymbol{T}_n(\boldsymbol{a}\otimes\boldsymbol{I}_3) + (\boldsymbol{a}\otimes\boldsymbol{I}_3)^H\boldsymbol{T}_n(\bar{\boldsymbol{a}}\otimes\boldsymbol{I}_3)\\ + &\ (\bar{\boldsymbol{a}}\otimes\boldsymbol{I}_3)^H\boldsymbol{T}_n(\bar{\boldsymbol{a}}\otimes\boldsymbol{I}_3)\end{aligned} \tag{12}$$

Define $D = C^{y\Delta x + x\Delta y + \Delta x\Delta y}$. As $\Delta x, \Delta y \to 0$, we have $D \to 1$. By (5), the elements $\bar{A}_{ij}, i, j \in [1,3]$ can be expressed in the following form (ignoring the constant factor determined by i, j)

$$\begin{aligned}\bar{A}_{ij} \ \leftrightarrow &\ \sum_{m=1}^{N-1}\sum_{n=1}^{N-1}(D^m-1)^H(D^n-1)\\ + &\ \sum_{m=1}^{N-1}(D^m-1)^H + \sum_{n=1}^{N-1}(D^n-1)\\ = &\ O(E) + O(E^2)\end{aligned} \tag{13}$$

The symbol 'O' denotes the infinitesimal order. Ignoring the high order infinitesimal, we have

$$\bar{A}_{ij} = O(E) \tag{14}$$

Assume the original matrix $\boldsymbol{A}$ is in the form

$$\boldsymbol{A} = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} \tag{15}$$

Here we take the first cofactor of $\boldsymbol{A}$ as an example, i.e.,

$$\boldsymbol{A}_{1,1} = \begin{bmatrix} A_{22} & A_{23} \\ A_{32} & A_{33} \end{bmatrix} \tag{16}$$

By adding the cofactor of difference matrix $\bar{\boldsymbol{A}}_{1,1}$, we have

$$\hat{\boldsymbol{A}}_{1,1} = \begin{bmatrix} A_{22} + \bar{A}_{22} & A_{23} + \bar{A}_{23} \\ A_{32} + \bar{A}_{32} & A_{33} + \bar{A}_{33} \end{bmatrix} \tag{17}$$

The determinant is given by

$$\begin{aligned} \mathbf{det}\{\hat{\boldsymbol{A}}_{1,1}\} &= \bar{A}_{22}\bar{A}_{33} - \bar{A}_{23}\bar{A}_{32} + A_{22}\bar{A}_{33} \\ &\quad + A_{33}\bar{A}_{22} - A_{23}\bar{A}_{32} - A_{32}\bar{A}_{23} \\ &= O(E) \end{aligned} \tag{18}$$

Similar with other cofactor matrices, the infinitesimal order of the cofactor matrix $\mathbf{det}\{\hat{\boldsymbol{A}}_{i,j}\}$ will be

$$\mathbf{det}\{\hat{\boldsymbol{A}}_{i,j}\} = O(E) \tag{19}$$

From (8), the relationship between the practical $\hat{\boldsymbol{A}}$ and $\hat{\boldsymbol{B}}$ can be denoted as

$$\begin{aligned} \hat{\boldsymbol{B}} &= \hat{\boldsymbol{\Omega}}^H \hat{\boldsymbol{A}} \hat{\boldsymbol{\Omega}} \\ &= \boldsymbol{\Omega}^H \boldsymbol{A} \boldsymbol{\Omega} + \bar{\boldsymbol{\Omega}}^H \boldsymbol{A} \boldsymbol{\Omega} + \boldsymbol{\Omega}^H \boldsymbol{A} \bar{\boldsymbol{\Omega}} + \bar{\boldsymbol{\Omega}}^H \boldsymbol{A} \bar{\boldsymbol{\Omega}} \\ &+ \boldsymbol{\Omega}^H \bar{\boldsymbol{A}} \boldsymbol{\Omega} + \bar{\boldsymbol{\Omega}}^H \bar{\boldsymbol{A}} \boldsymbol{\Omega} + \boldsymbol{\Omega}^H \bar{\boldsymbol{A}} \bar{\boldsymbol{\Omega}} + \bar{\boldsymbol{\Omega}}^H \bar{\boldsymbol{A}} \bar{\boldsymbol{\Omega}} \\ &= \boldsymbol{B} + \bar{\boldsymbol{B}} \end{aligned} \tag{20}$$

where

$$\begin{aligned} \bar{\boldsymbol{B}} &= \bar{\boldsymbol{\Omega}}^H \boldsymbol{A} \boldsymbol{\Omega} + \boldsymbol{\Omega}^H \boldsymbol{A} \bar{\boldsymbol{\Omega}} + \bar{\boldsymbol{\Omega}}^H \boldsymbol{A} \bar{\boldsymbol{\Omega}} + \boldsymbol{\Omega}^H \bar{\boldsymbol{A}} \boldsymbol{\Omega} \\ &+ \bar{\boldsymbol{\Omega}}^H \bar{\boldsymbol{A}} \boldsymbol{\Omega} + \boldsymbol{\Omega}^H \bar{\boldsymbol{A}} \bar{\boldsymbol{\Omega}} + \bar{\boldsymbol{\Omega}}^H \bar{\boldsymbol{A}} \bar{\boldsymbol{\Omega}} \end{aligned} \tag{21}$$

It can be obtained that the elements in $\bar{\boldsymbol{B}}$ consist of the linear combination of infinitesimals $\bar{\Omega}_{ij}^H$, $\bar{\Omega}_{ij}$, $\bar{\Omega}_{ij}^H\bar{\Omega}_{ij}$, $\bar{A}_{ij}$ $\bar{\Omega}_{ij}^H\bar{A}_{ij}$, $\bar{A}_{ij}\bar{\Omega}_{ij}$ and $\bar{\Omega}_{ij}^H\bar{A}_{ij}\bar{\Omega}_{ij}$. As discussed above, $\bar{A}_{ij} = O(E)$. It can be verified that, for all infinitesimals $\bar{\Omega}_{ij}$, we have

$$\frac{\bar{A}_{ij}}{\bar{\Omega}_{ij}} \to \infty \tag{22}$$

which means $\bar{\Omega}_{ij}$ has higher infinitesimal order than $O(E)$. Keeping the lowest order of infinitesimal in $\bar{\boldsymbol{B}}$, the elements has the order

$$\bar{B}_{ij} = O(E) \tag{23}$$

As DST signals have $\mathbf{rank}\{\boldsymbol{B}\} = 0$, then $\boldsymbol{B}$ is in the form

$$\boldsymbol{B} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \tag{24}$$

The determinant of $\hat{\boldsymbol{B}}$ is denoted by

$$\mathbf{det}\{\hat{\boldsymbol{B}}\} \;=\; \bar{B}_{11}\bar{B}_{22} - \bar{B}_{12}\bar{B}_{21} \tag{25}$$

The infinitesimal order is

$$\mathbf{det}\{\hat{\boldsymbol{B}}\} = O(E^2) \tag{26}$$

The estimator is calculated as

$$\begin{aligned} F_3(x+\Delta x, y+\Delta y) &= \frac{\sum_{i=1}^{3}\sum_{j=1}^{3}\mathbf{det}\{\hat{\boldsymbol{A}}_{i,j}\}}{\mathbf{det}\{\hat{\boldsymbol{B}}\}} \\ &= \frac{O(E)}{O(E^2)} \to \infty \end{aligned} \tag{27}$$

The infinity value indicates the peaks in the DOA spectrum and the estimator could also find the DST signal directions successfully.

Case 3: For DST ambiguity directions, $\mathbf{rank}\{A\} = 1$ and $\mathbf{rank}\{B\} = 1$. The numerator of the estimator is the same with the DST signal direction case, which is denoted by

$$\sum_{i=1}^{3}\sum_{j=1}^{3}\mathbf{det}\{\hat{\boldsymbol{A}}_{i,j}\} = O(E) \tag{28}$$

However, the rank of theoretical matrix $\boldsymbol{B}$ equals 1 instead of 0, which means $\boldsymbol{B}$ is not a zero matrix and can't be ignored. Then we have

$$\begin{aligned}\hat{\boldsymbol{B}} &= \boldsymbol{B} + \bar{\boldsymbol{B}} \\ &= \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix} + \begin{bmatrix} \bar{B}_{11} & \bar{B}_{12} \\ \bar{B}_{21} & \bar{B}_{22} \end{bmatrix} \\ &= \begin{bmatrix} \bar{B}_{11} + B_{11} & \bar{B}_{12} + B_{12} \\ \bar{B}_{21} + B_{21} & \bar{B}_{22} + B_{22} \end{bmatrix} \end{aligned} \tag{29}$$

where $B_{11}, B_{12}, B_{21}, B_{22}$ are constants which cannot equal to zero simultaneously. The denominator of the estimator can be calculated as

$$\begin{aligned}\mathbf{det}\{\hat{\boldsymbol{B}}\} &= \bar{B}_{11}\bar{B}_{22} - \bar{B}_{12}\bar{B}_{21} + B_{22}\bar{B}_{11} \\ &\quad + B_{11}\bar{B}_{22} - B_{21}\bar{B}_{12} - B_{12}\bar{B}_{21} \\ &= O(E) \end{aligned} \tag{30}$$

In this case, the estimator is in the form of

$$F_3(x + \Delta x, y + \Delta y) = \frac{O(E)}{O(E)} \tag{31}$$

The final results will approach an undetermined constants instead of infinity. In the DOA spectrum, the DST ambiguity directions will not appear as a peak.

Case 4: For the uninterested directions, $\mathbf{rank}\{A\} = 3$ and $\mathbf{rank}\{B\} = 2$, which means these two matrices are full-rank matrices. The numerator of the estimator $\sum_{i=1}^{3}\sum_{j=1}^{3} \mathbf{det}\{\hat{\boldsymbol{A}}_{i,j}\}$ will be a non-zero constant, and so is the denominator $\mathbf{det}\{\hat{\boldsymbol{B}}\}$. The results of the estimator are finite values and these directions will not appear as peaks in the DOA spectrum.

Bibliography

[1] J. Schweitzer, J. Fyen, S. Mykkeltveit, T. Kværna, and P. Bormann, "Seismic arrays," *IASPEI new manual of seismological observatory practice*, p. 52, 2002.

[2] J. M. Cannata, J. A. Williams, Q. Zhou, T. A. Ritter, and K. K. Shung, "Development of a 35-mhz piezo-composite ultrasound array for medical imaging," *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, vol. 53, no. 1, pp. 224–236, 2006.

[3] F. Vignon and M. R. Burcher, "Capon beamforming in medical ultrasound imaging with focused beams," *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, vol. 55, no. 3, pp. 619–628, 2008.

[4] B. D. Steinberg, "Microwave imaging with large antenna arrays: radio camera principles and techniques," *New York, Wiley-Interscience, 1983, 323 p.*, 1983.

[5] D. M. Green, J. A. Swets *et al.*, *Signal detection theory and psychophysics*. Wiley New York, 1966, vol. 1.

[6] J. Li and P. Stoica, Eds., *Robust Adaptive Beamforming*. Hoboken, New Jersey: John Wiley & Sons, 2005.

[7] B. D. Van Veen and K. M. Buckley, "Beamforming: A versatile approach to spatial filtering," *IEEE assp magazine*, vol. 5, no. 2, pp. 4–24, 1988.

[8] P. Ioannides and C. A. Balanis, "Uniform circular and rectangular arrays for adaptive

beamforming applications," *IEEE Antennas and Wireless Propagation Letters*, vol. 4, pp. 351–354, 2005.

[9] S.-S. Jeon, Y. Wang, Y. Qian, and T. Itoh, "A novel planar array smart antenna system with hybrid analog-digital beamforming," in *2001 IEEE MTT-S International Microwave Sympsoium Digest (Cat. No. 01CH37157)*, vol. 1. IEEE, 2001, pp. 121–124.

[10] B.-H. Kim, T.-K. Song, Y. Yoo, J. H. Chang, S. Lee, Y. Kim, K. Cho, and J. Song, "Hybrid volume beamforming for 3-d ultrasound imaging using 2-d cmut arrays," in *2012 IEEE International Ultrasonics Symposium*. IEEE, 2012, pp. 2246–2249.

[11] I. Bekkerman and J. Tabrikian, "Target detection and localization using mimo radars and sonars," *IEEE Transactions on Signal Processing*, vol. 54, no. 10, pp. 3873–3883, 2006.

[12] L. Wan, W. Si, L. Liu, Z. Tian, and N. Feng, "High accuracy 2d-doa estimation for conformal array using parafac," *International Journal of Antennas and Propagation*, vol. 2014, 2014.

[13] L. C. Godara, "Application of antenna arrays to mobile communications. ii. beamforming and direction-of-arrival considerations," *Proceedings of the IEEE*, vol. 85, no. 8, pp. 1195–1245, August 1997.

[14] G. A. Rusnak, "Beamforming in seismic surveying," Nov. 18 1969, uS Patent 3,479,638.

[15] P. S. Naidu, *Sensor array signal processing*. CRC press, 2009.

[16] R. Compton Jr, "On the performance of a polarization sensitive adaptive array," *IEEE Transactions on Antennas and Propagation*, vol. 29, no. 5, pp. 718–725, September 1981.

[17] M. M. Goodwin and G. W. Elko, "Constant beamwidth beamforming," in *1993 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 1. IEEE, 1993, pp. 169–172.

[18] J. Capon, "High-resolution frequency-wavenumber spectrum analysis," *Proceedings of the IEEE*, vol. 57, no. 8, pp. 1408–1418, August 1969.

[19] J. Li, P. Stoica, and Z. Wang, "On robust capon beamforming and diagonal loading," *IEEE transactions on signal processing*, vol. 51, no. 7, pp. 1702–1715, 2003.

[20] L. Zhang and W. Liu, "Robust beamforming for coherent signals based on the spatial-smoothing technique," *Signal Processing*, vol. 92, no. 11, pp. 2747–2758, 2012.

[21] L. Yu, W. Liu, and R. J. Langley, "Robust beamforming methods for multipath signal reception," *Digital Signal Processing*, vol. 20, no. 2, pp. 379–390, 2010.

[22] S. A. Vorobyov, A. B. Gershman, and Z.-Q. Luo, "Robust adaptive beamforming using worst-case performance optimization: A solution to the signal mismatch problem," *IEEE transactions on signal processing*, vol. 51, no. 2, pp. 313–324, 2003.

[23] J. Huang and A. L. Swindlehurst, "Robust secure transmission in miso channels based on worst-case optimization," *IEEE Transactions on Signal Processing*, vol. 60, no. 4, pp. 1696–1707, 2012.

[24] J. Wang and D. P. Palomar, "Worst-case robust mimo transmission with imperfect channel knowledge," *IEEE Transactions on Signal Processing*, vol. 57, no. 8, pp. 3086–3100, 2009.

[25] S.-J. Kim, A. Magnani, A. Mutapcic, S. P. Boyd, and Z.-Q. Luo, "Robust beamforming via worst-case sinr maximization," *IEEE Transactions on Signal Processing*, vol. 56, no. 4, pp. 1539–1547, 2008.

[26] R. O. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Transactions on Antennas and Propagation*, vol. 34, no. 3, pp. 276–280, March 1986.

[27] R. Roy and T. Kailath, "ESPRIT-estimation of signal parameters via rotational invariance techniques," *IEEE Transactions on acoustics, speech, and signal processing*, vol. 37, no. 7, pp. 984–995, 1989.

[28] R. J. Weber and Y. Huang, "Analysis for capon and music doa estimation algorithms," in *2009 IEEE Antennas and Propagation Society International Symposium*. IEEE, 2009, pp. 1–4.

[29] W. Liu, "Design and implementation of a rectangular frequency invariant beamformer with a full azimuth angle coverage," *Journal of the Franklin Institute*, vol. 348, no. 9, pp. 2556–2569, 2011.

[30] F.-J. Chen, S. Kwong, and C.-W. Kok, "Esprit-like two-dimensional doa estimation for coherent signals," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 46, no. 3, pp. 1477–1484, 2010.

[31] W. Zhang, W. Liu, J. Wang, and S. Wu, "Computationally efficient 2-d doa estimation for uniform rectangular arrays," *Multidimensional Systems and Signal Processing*, vol. 25, no. 4, pp. 847–857, 2014.

[32] M. Agatonovic, Z. Stanković, and B. Milovanović, "High resolution two-dimensional doa estimation using artificial neural networks," in *2012 6th European Conference on Antennas and Propagation (EUCAP)*. IEEE, 2012, pp. 1–5.

[33] A. Nehorai and E. Paldi, "Acoustic vector-sensor array processing," *IEEE Transactions on signal processing*, vol. 42, no. 9, pp. 2481–2491, 1994.

[34] ——, "Vector-sensor array processing for electromagnetic source localization," *IEEE transactions on signal processing*, vol. 42, no. 2, pp. 376–398, 1994.

[35] A. J. Weiss and B. Friedlander, "Maximum likelihood signal estimation for polarization sensitive arrays," *IEEE Transactions on Antennas and Propagation*, vol. 41, no. 7, pp. 918–925, 1993.

[36] S. Miron, N. Le Bihan, and J. I. Mars, "Vector-sensor music for polarized seismic sources localization," *EURASIP Journal on Applied Signal Processing*, vol. 2005, pp. 74–84, 2005.

[37] W. Guo, M. Yang, B. Chen, and G. Zheng, "Joint doa and polarization estimation using music method in polarimetric mimo radar," 2012.

[38] L. Wang, L. Yang, G. Wang, Z. Chen, and M. Zou, "Uni-vector-sensor dimensionality reduction music algorithm for doa and polarization estimation," *Mathematical Problems in Engineering*, vol. 2014, 2014.

[39] A. Nehorai, K. C. Ho, and B. T. G. Tan, "Minimum-noise-variance beamformer with an electromagnetic vector sensor," *IEEE Transactions on Signal Processing*, vol. 47, no. 3, pp. 601–618, March 1999.

[40] Y. G. Xu, T. Liu, and Z. W. Liu, "Output SINR of MV beamformer with one EM vector sensor of and magnetic noise power," in *Proc. of International Conference on Signal Processing*, September 2004, pp. 419–422.

[41] N. Le Bihan and J. Mars, "Singular value decomposition of quaternion matrices: a new tool for vector-sensor signal processing," *Signal Processing*, vol. 84, no. 7, pp. 1177–1199, 2004.

[42] S. Miron, N. Le Bihan, and J. I. Mars, "High resolution vector-sensor array processing using quaternions," in *Proc. of IEEE Workshop on Statistical Signal Processing*, July 2005, pp. 918–923.

[43] Miron, Sebastian and Le Bihan, Nicolas and Mars, Jerome I, "Quaternion-MUSIC for vector-sensor array processing," *IEEE Transactions on Signal Processing*, vol. 54, no. 4, pp. 1218–1229, April 2006.

[44] X. Gong, Y. Xu, and Z. Liu, "Quaternion ESPRIT for direction finding with a

polarization sentive array," in *Proc. International Conference on Signal Processing*, 2008, pp. 378–381.

[45] J. W. Tao and W. X. Chang, "A novel combined beamformer based on hypercomplex processes," *taes*, vol. 49, no. 2, pp. 1276–1289, 2013.

[46] J.-W. Tao, "Performance analysis for interference and noise canceller based on hypercomplex and spatio-temporal-polarisation processes," *IET Radar, Sonar Navigation*, vol. 7, no. 3, pp. 277–286, 2013.

[47] X. R. Zhang, W. Liu, Y. G. Xu, and Z. W. Liu, "Quaternion-valued robust adaptive beamformer for electromagnetic vector-sensor arrays with worst-case constraint," *Signal Processing*, vol. 104, pp. 274–283, November 2014.

[48] M. D. Jiang, W. Liu, and Y. Li, "Adaptive beamforming for vector-sensor arrays based on reweighted zero-attracting quaternion-valued LMS algorithm," *IEEE Trans. on Circuits and Systems II: Express Briefs*, vol. 63, pp. 274–278, March 2016.

[49] X. Gou, Y. Xu, Z. Liu, and X. Gong, "Quaternion-capon beamformer using crossed-dipole arrays," in *Proc. of IEEE International Symposium on Microwave, Antenna, Propagation, and EMC Technologies for Wireless Communications (MAPE)*, November 2011, pp. 34–37.

[50] O. M. Isaeva and V. A. Sarytchev, "Quaternion presentations polarization state," in *Proc. 2nd IEEE Topical Symposium of Combined Optical-Microwave Earth and Atmosphere Sensing*, Atlanta, US, April 1995, pp. 195–196.

[51] W. Liu, "Channel equalization and beamforming for quaternion-valued wireless communication systems," *Journal of the Franklin Institute*, DOI: 10.1016/j.jfranklin.2016.10.043, 2016.

[52] W. Liu and S. Weiss, *Wideband Beamforming: Concepts and Techniques*. Chichester, UK: John Wiley & Sons, 2010.

[53] S. A. Vorobyov, A. B. Gershman, and Z. Q. Luo, "Robust adaptive beamforming using worst-case performance optimization: A solution to the signal mismatch problem," *IEEE Transactions on Signal Processing*, vol. 51, no. 2, pp. 313–324, February 2003.

[54] L. Yu, W. Liu, and R. J. Langley, "Novel robust beamformers for coherent interference suppression with DOA estimation errors," *IET Microwaves, Antennas and Propagation*, vol. 4, pp. 1310–1319, September 2010.

[55] W. R. Hamilton, "Ii. on quaternions; or on a new system of imaginaries in algebra," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 25, no. 163, pp. 10–13, July 1844.

[56] L. Huang and W. So, "On left eigenvalues of a quaternionic matrix," *Linear algebra and its applications*, vol. 323, no. 1, pp. 105–116, January 2001.

[57] F. Zhang, "Quaternions and matrices of quaternions," *Linear algebra and its applications*, vol. 251, pp. 21–57, January 1997.

[58] M. D. Jiang, Y. Li, and W. Liu, "Properties of a general quaternion-valued gradient operator and its application to signal processing," *Frontiers of Information Technology & Electronic Engineering*, vol. 17, pp. 83–95, February 2016.

[59] O. L. Frost, III, "An algorithm for linearly constrained adaptive array processing," *Proceedings of the IEEE*, vol. 60, no. 8, pp. 926–935, August 1972.

[60] M. Wang and W. Ma, "A structure-preserving algorithm for the quaternion cholesky decomposition," *Applied Mathematics and Computation*, vol. 223, pp. 354–361, 2013.

[61] Y. Xu and Z. Liu, "Simultaneous estimation of 2-D DOA and polarization of multiple coherent sources using an electromagnetic vector sensor array," *Journal-China Institute of Communications*, vol. 25, no. 5, pp. 28–38, 2004.

[62] S. Liu, M. Jin, and X. Qiao, "Joint polarization-DOA estimation using circle array," in *Proc. IET International Radar Conference*, 2009, pp. 1–5.

[63] M. Costa, A. Richter, and V. Koivunen, "DOA and polarization estimation for arbitrary array configurations," *IEEE Transactions on Signal Processing*, vol. 60, no. 5, pp. 2330–2343, 2012.

[64] X. R. Zhang, Z. W. Liu, W. Liu, and Y. G. Xu, "Quasi-vector-cross-product based direction finding algorithm with a spatially stretched tripole," in *Proc. of the IEEE TENCON Conference*, Xi'an, China, October 2013.

[65] X. Yuan, K. T. Wong, Z. Xu, and K. Agrawal, "Various compositions to form a triad of collocated dipoles/loops, for direction finding and polarization estimation," *IEEE sensors Journal*, vol. 12, no. 6, pp. 1763–1771, 2012.

[66] M. D. Zoltowski and K. T. Wong, "Esprit-based 2-d direction finding with a sparse uniform array of electromagnetic vector sensors," *IEEE Transactions on Signal Processing*, vol. 48, no. 8, pp. 2195–2204, 2000.

[67] X. Zhang, C. Chen, J. Li, and D. Xu, "Blind DOA and polarization estimation for polarization-sensitive array using dimension reduction MUSIC," *Multidimensional Systems and Signal Processing*, vol. 25, no. 1, pp. 67–82, 2014.

[68] Y. Hua, "A pencil-MUSIC algorithm for finding two-dimensional angles and polarizations using crossed dipoles," *IEEE Transactions on Antennas and Propagation*, vol. 41, no. 3, pp. 370–376, 1993.

[69] J. Lundback and S. Nordebo, "Analysis of a tripole array for polarization and direction of arrival estimation," in *Proc. IEEE Workshop on Sensor Array and Multichannel Signal Processing*, July 2004, pp. 284–288.

[70] M. B. Hawes, W. Liu, and L. Mihaylova, "Compressive sensing based design of sparse tripole arrays," *Sensors*, vol. 15, no. 12, pp. 31 056–31 068, 2015.

[71] K. Wang, J. He, T. Shu, and Z. Liu, "Angle-polarization estimation for coherent sources with linear tripole sensor arrays," *Sensors*, vol. 16, no. 2, p. 248, 2016.

[72] X. Lan, W. Liu, and H. Y. Ngan, "Joint 4-D DOA and polarization estimation based on linear tripole arrays," in *Digital Signal Processing (DSP), 2017 22nd International Conference on*. IEEE, 2017, pp. 1–5.

[73] H. L. Van Trees, *Detection, estimation, and modulation theory, optimum array processing*. John Wiley & Sons, 2004.

[74] P. Stoica and A. Nehorai, "MUSIC, maximum likelihood, and Cramer-Rao bound," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 5, pp. 720–741, 1989.

[75] P. Stoica and K. C. Sharman, "Maximum likelihood methods for direction-of-arrival estimation," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 38, no. 7, pp. 1132–1143, 1990.

[76] Y. Bresler and A. Macovski, "Exact maximum likelihood parameter estimation of superimposed exponential signals in noise," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 34, no. 5, pp. 1081–1089, 1986.

[77] H. Abeida and J. P. Delmas, "Direct derivation of the stochastic CRB of DOA estimation for rectilinear sources," *IEEE Signal Processing Letters*, vol. 24, no. 10, pp. 1522–1526, 2017.

[78] L. Kumar and R. M. Hegde, "Stochastic Cramér-Rao bound analysis for DOA estimation in spherical harmonics domain," *IEEE Signal Processing Letters*, vol. 22, no. 8, pp. 1030–1034, 2015.

[79] W. Lv, H. Sun, X. Zhang, and D. Xu, "Reduced-dimension noncircular-capon algorithm for DOA estimation of noncircular signals," *International Journal of Antennas and Propagation*, vol. 2015, 2015.

[80] M. Jin, G. Liao, and J. Li, "Joint DOD and DOA estimation for bistatic MIMO radar," *Signal Processing*, vol. 89, no. 2, pp. 244–251, 2009.

[81] F. Roemer and M. Haardt, "Efficient 1-D and 2-D DOA estimation for non-circular sources with hexagonal shaped ESPAR arrays," in *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, vol. 4. IEEE, 2006, pp. IV–IV.

[82] P. Stoica, E. G. Larsson, and A. B. Gershman, "The stochastic CRB for array processing: A textbook derivation," *IEEE Signal Processing Letters*, vol. 8, no. 5, pp. 148–150, 2001.

[83] A. Nehorai and E. Paldi, "Electromagnetic vector-sensor array processing," in *Digital Signal Processing Handbook*, 1998, pp. 65–1.

[84] H. Roger and R. J. Charles, "Topics in matrix analysis," 1994.

[85] P. Stoica and A. Nehorai, "Performance study of conditional and unconditional direction-of-arrival estimation," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 38, no. 10, pp. 1783–1795, 1990.

[86] H. Chen, C. Hou, W. Liu, W.-P. Zhu, and M. Swamy, "Efficient two-dimensional direction-of-arrival estimation for a mixture of circular and noncircular sources," *IEEE Sensors Journal*, vol. 16, no. 8, pp. 2527–2536, 2016.

[87] A. B. Gershman, M. Rübsamen, and M. Pesavento, "One-and two-dimensional direction-of-arrival estimation: An overview of search-free techniques," *Signal Processing*, vol. 90, no. 5, pp. 1338–1349, 2010.

[88] B. Liao, A. Madanayake, and P. Agathoklis, "Array signal processing and systems," *Multidimensional Systems and Signal Processing*, vol. 29, no. 2, pp. 467–473, 2018.

[89] J. M. Kim, O. K. Lee, and J. C. Ye, "Compressive MUSIC: Revisiting the link

between compressive sensing and array signal processing," *IEEE Transactions on Information Theory*, vol. 58, no. 1, pp. 278–301, 2012.

[90] Q. Shen, W. Liu, W. Cui, and S. Wu, "Underdetermined DOA estimation under the compressive sensing framework: A review," *IEEE Access*, vol. 4, pp. 8865–8878, 2016.

[91] Z. Xin, S. Yaowu, and Y. Wenhong, "2-D DOA and polarization estimation of LFM signals with one electromagnetic vector sensor," in *Signal Processing, 2008. ICSP 2008. 9th International Conference on.* IEEE, 2008, pp. 386–389.

[92] K. T. Wong and M. D. Zoltowski, "Uni-vector-sensor esprit for multisource azimuth, elevation, and polarization estimation," *IEEE Transactions on Antennas and Propagation*, vol. 45, no. 10, pp. 1467–1474, 1997.

[93] X. Zhang, Y. Shi, and D. Xu, "Novel blind joint direction of arrival and polarization estimation for polarization-sensitive uniform circular array," *Progress In Electromagnetics Research*, vol. 86, pp. 19–37, 2008.

[94] A. Richter, F. Belloni, and V. Koivunen, "Doa and polarization estimation using arbitrary polarimetric array configurations," in *Fourth IEEE Workshop on Sensor Array and Multichannel Processing, 2006.* IEEE, 2006, pp. 55–59.

[95] J. Li and R. Compton, "Two-dimensional angle and polarization estimation using the esprit algorithm," *IEEE Transactions on Antennas and Propagation*, vol. 40, no. 5, pp. 550–555, 1992.

[96] A. Nehorai and E. Paldi, "Electromagnetic vector-sensor array processing," pp. 65–1, 1998.

[97] S. S. Qureshi, S. Ali, and S. A. Hassan, "Optimal polarization diversity gain in dual-polarized antennas using quaternions," *IEEE Signal Processing Letters*, vol. 25, no. 4, pp. 467–471, 2018.

[98] W. Liu, "Channel equalization and beamforming for quaternion-valued wireless communication systems," *Journal of the Franklin Institute*, vol. 354, no. 18, pp. 8721–8733, 2017.