New Directions In Sparse Sampling and Estimation For Underdetermined Systems

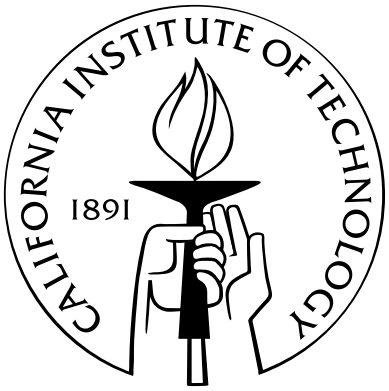
Thesis by

Piya Pal

In Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy



California Institute of Technology

Pasadena, California

2013

(Defended June 3rd, 2013)

To Maa, Baba, and my Brother

Acknowledgments

My years as a graduate student at Caltech have been one of the most defining phases of my life, one in which I grew both professionally and as an individual. As I fondly reflect back upon these years, I would like to thank many who have contributed in more than one way to this journey.

I have been most fortunate to have Prof. P. P. Vaidyanathan as my adviser. I am deeply indebted to him for giving me the freedom to pursue my ideas in research and providing unconditional support at every stage. He has inspired me to put utmost emphasis on creativity and originality, even if it requires one to boldly challenge decade-old established ideas. He has encouraged me to blaze new trails, rather than to follow an established one; to keep persevering even when the initial outcomes look disappointing; to be my own harshest critic so that I "bend over my back" to point out any potential drawback of my work; and above all, to enjoy this process every single day of my life. His life is an embodiment of work ethics and I have learned a lot by merely observing him on a daily basis. Finally, it is a pleasant coincidence that we share the same initials (P. P.) and it serves as a constant reminder of the great responsibility that comes with it.

My gratitude goes to my thesis committee: Prof. Babak Hassibi, Prof. Yaser Abu Mostafa, Dr. Jakob Van Zyl, Dr. Andre Tkacenko, and Prof. Shuki Bruck who was unavailable for my defense but had served on my candidacy committee. Babak has been a constant source of support and encouragement ever since my first year at Caltech. From him, I have received invaluable advice regarding career and professional choices. I have also greatly benefited from our technical discussions. Even a brief conversation with him has often inspired a new direction of thinking. I am indebted to Yaser for his words of encouragement. I also learned a lot from our technical discussions where he provided valuable insights and alternate viewpoints. My interactions with Jakob have given rise to several ideas that I wish to pursue as my future course of research, especially those pertaining to radar and applications in remote sensing. As a past graduate student of our lab, Andre has been a great friend and almost a big brother to me. I greatly value our conversations

on various topics ranging from academics to life in general. I would also like to express my deepest gratitude to Shuki, who has been a source of great support and inspiration, ever since my very first year at Caltech.

I would like to gratefully acknowledge the support of the Office of Naval Research (ONR) and Caltech which made this work possible.

Throughout my graduate years at Caltech, I have immensely benefited from interactions with some of the brightest minds. My special thanks extend out to the past and present members of my group, including Prof. Byung-Jun Yoon, Dr. Chun-Yang Chen (Scott), Dr. Ching-Chih Weng (Brian), Dr. Chih-Hao Liu (John), and Srikanth Tenneti. I learned a lot about life as a Ph. D student by observing Scott and Brian during my formative years. Although Byung-Jun had already left the group by the time I joined, he was always there when I needed him and provided generous advice regarding my career path. John has been my office mate and a dear friend for almost 5 years - we have shared some great times together, including conference trips to Paris and Prague, as well as our annual drive to Asilomar.

As I leave Caltech for the next phase of my career, I will be carrying fond memories of wonderful moments spent with some of the greatest friends I have. I would like to thank Teja, Rangoli, Prakhar, Nakul, JK, Sushree, Srivatsan, Vikas, Abhishek Saha, Kaushik Sengupta, Wei Mao, Na Li, Elizabeth and Zhiying for their friendship and company. I would also like to thank my friends among the staff in various offices in Caltech. I am thankful to Andrea Boyle, Caroline Murphy, Tanya Owen, and Lucinda Acosta for the care and promptness with which they took care of administrative matters. The international Students Program (ISP) Office at Caltech has always been a special place for me. It was at ISP where I met with many of my friends for the first time and I have always enjoyed being associated with ISP ever since. My special thanks to Laura Flower Kim and Daniel Yoder for their support and friendship. Laura has been a dear friend and I am going to treasure my many conversations with her. I would also like to express my sincere gratitude to Dr. Felicia Hunt - I cannot thank her enough for being a constant source of support and encouragement during my years at Caltech. The Caltech Glee Club has been an integral part of my life which I am going to dearly miss. I would like to thank our wonderful instructor Nancy Sulahian as well as dear friends Soyoung Park and Tiffany Kim for the great time we spend together on Monday and Wednesday evenings.

It is hard to find words to describe what my father Debabrata Pal and mother Malobika Pal

contributed towards making me what I am today. Without their unconditional love and many sacrifices, I would not have come this far. My dad has always inspired me to boldly take up challenges in life and not be deterred by them, leading largely by his own example. I owe my love for literature, art and music to my mom, who is a voracious reader with razor sharp memory. My brother, Subhabrata Pal, inspite of being a couple of years younger to me, has always acted like the ever-protective "big brother" I never had! My thesis is dedicated to my family.

Abstract

A central objective in signal processing is to infer meaningful information from a set of measurements or data. While most signal models have an overdetermined structure (the number of unknowns less than the number of equations), traditionally very few statistical estimation problems have considered a data model which is underdetermined (number of unknowns more than the number of equations). However, in recent times, an explosion of theoretical and computational methods have been developed primarily to study underdetermined systems by imposing sparsity on the unknown variables. This is motivated by the observation that inspite of the huge volume of data that arises in sensor networks, genomics, imaging, particle physics, web search etc., their information content is often much smaller compared to the number of raw measurements. This has given rise to the possibility of reducing the number of measurements by down sampling the data, which automatically gives rise to underdetermined systems.

In this thesis, we provide new directions for estimation in an underdetermined system, both for a class of parameter estimation problems and also for the problem of sparse recovery in compressive sensing. There are two main contributions of the thesis: design of new sampling and statistical estimation algorithms for array processing, and development of improved guarantees for sparse reconstruction by introducing a statistical framework to the recovery problem.

We consider underdetermined observation models in array processing where the number of unknown sources simultaneously received by the array can be considerably larger than the number of physical sensors. We study new sparse spatial sampling schemes (array geometries) as well as propose new recovery algorithms that can exploit priors on the unknown signals and unambiguously identify all the sources. The proposed sampling structure is generic enough to be extended to multiple dimensions as well as to exploit different kinds of priors in the model such as correlation, higher order moments, etc.

Recognizing the role of correlation priors and suitable sampling schemes for underdetermined

estimation in array processing, we introduce a correlation aware framework for recovering sparse support in compressive sensing. We show that it is possible to strictly increase the size of the recoverable sparse support using this framework provided the measurement matrix is suitably designed. The proposed nested and coprime arrays are shown to be appropriate candidates in this regard. We also provide new guarantees for convex and greedy formulations of the support recovery problem and demonstrate that it is possible to strictly improve upon existing guarantees.

This new paradigm of underdetermined estimation that explicitly establishes the fundamental interplay between sampling, statistical priors and the underlying sparsity, leads to exciting future research directions in a variety of application areas, and also gives rise to new questions that can lead to stand-alone theoretical results in their own right.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The theory of statistical estimation is concerned with extraction of meaningful information from available data or measurements. In recent times, there has been an exponential rate of increase in the amount of data that is available in almost every application area, leading to the so called "data deluge" [8]. The "Big Data" routinely arises in sensor networks, genomics, remote sensing, imaging, particle physics, web search, social networks, and so forth. This has led to a widening gap between the volume of available data and the capabilities for storing, communicating and processing them efficiently. Fortunately however, the amount of information buried in the data acquired by sensors in most scenarios is substantially lower compared to the number of raw samples acquired. This key observation has led to the possibility of sampling strategies and design of sensing systems that can directly capture the information using far fewer samples (typically achieved by random projections). In many natural scenarios, the physics of the problem itself imposes structures on the ensuing acquisition system, leading to the possibility of "structured sampling" strategies. Also often, one can make informed assumptions about the nature of randomness, or statistical distribution of the data which can be possibly exploited to our advantage for extracting information. In fact, in many cases, one is actually interested in estimating parameters of the underlying distribution of the model from the statistics of the data.

Our research aims at understanding the role of **sparse structured sampling** in the extraction of information (in the form of parameters of the distribution) buried in data, and investigating if such sampling can directly exploit the **statistics** of the signal. Can assumptions on an underlying statistical model be leveraged in our sampling strategies to obtain only a few samples from the signals of interest without loss of information? Also, can we design efficient algorithms to extract the desired information from these few judiciously captured samples, so that the computational

complexity does not drastically scale with the volume of the data?

The work presented in this thesis addresses these questions by revisiting a much unexplored frontier of estimation theory, namely that of an underdetermined observation model. The first part of the thesis focuses on underdetermined signal models arising in antenna array processing. It is devoted to development of new theoretical guarantees and practical sampling and estimation schemes to attain those guarantees. In the second part, we consider underdetermined systems which exhibit sparsity and demonstrate that our approach can be generalized to this case as well, leading to a new theory of correlation aware sparse estimation.

In this introductory chapter, we review the history and developments of the problems of interest in this thesis: namely algorithms for antenna array processing, and the theory of sparse signal representation and estimation. Due to the large volume of existing literature, the summary here is only directly related to the current thesis and is by no means a complete treatment of all past work. Please refer to [186, 104, 96, 187, 24, 56] for a more comprehensive treatment.

1.1 Estimation in Array Processing

We present a broad review of estimation algorithms used in array processing. Many of these techniques apply to more general estimation problems beyond the particular application of sensor arrays. The space-time model of array processing also has a more general interpretation which will be revisited in the context of sparse estimation and will be shown to find applications in many other problems.

1.1.1 Signal Model and Role of Autocorrelation

Array processing considers a space-time signal model where both spatial and temporal samples of signals are collected and processed simultaneously. The signal model is developed by considering both the physics of electromagnetic (EM) wave propagation as well as the statistical model of the impinging signals.

1.1.1.1 The Array Signal Model

Consider a plane wave $f(t;\mathbf{d})$ propagating in the direction pointed by the unit vector $\mathbf{v} \in \mathbb{R}^3$

$$\mathbf{v}(\theta,\phi) = \begin{bmatrix} \sin\theta\cos\phi \\ \sin\theta\sin\phi \\ \cos\theta \end{bmatrix} \tag{1.1}$$

Here θ denotes the elevation angle and ϕ denotes the azimuthal angle as represented in Fig. 1.1. Consider an array of M sensors where the position vector of the mth sensor is given by $\mathbf{d}_m \in \mathbb{R}^3$.

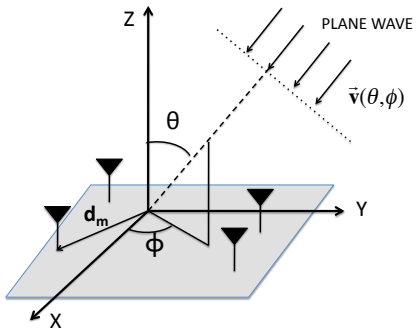


Figure 1.1: Figure showing plane wave propagating along the direction $\mathbf{v}(\theta,\phi)$ and received by an array of sensors. Here θ denotes the elevation and ϕ denotes the azimuthal angle.

The signal received at the M sensors can be represented as a vector

$$\mathbf{y}(t) = \begin{bmatrix} f(t-\tau_1) \\ f(t-\tau_2) \\ \vdots \\ f(t-\tau_M) \end{bmatrix}, \quad f(t-\tau_i) \triangleq f(t;\mathbf{d}_m), \quad \tau_m = \frac{1}{c}\mathbf{v}^T\mathbf{d}_m \tag{1.2}$$

where c is the velocity of propagation in the medium and $\mathbf{v}$ is as in (1.1). For a narrowband signal model, we can write

$$f(t;\mathbf{d}) = s(t;\mathbf{d})e^{j\omega_c t}$$

where ω_c represents the carrier frequency (in radians/s) and $s(t;\mathbf{d})$ is the complex envelope with a bandwidth of B_s. In array processing, a **narrowband** signal model is defined as one where

$$\max_m \tau_m \ll \frac{1}{B_s}$$

This is true if the bandwidth B_s of the envelope is considerably small. Under this assumption, $s(t-\tau_m) \approx s(t), \quad \forall m$ and (1.2) reduces to

$$\mathbf{y}(t) = \mathbf{a}(\theta,\phi)s(t)e^{j\omega_c t} \tag{1.3}$$

where $\mathbf{a}(\theta,\phi) \in \mathbb{C}^{M\times 1}$ is defined as the **steering vector** of the antenna array and its elements are given by $\frac{\omega_c}{c}\mathbf{v}^T\mathbf{d}_m, 1 \le m \le M$. Equation (1.3) is widely known as the narrowband signal model for far field wave propagation for a single source. For multiple plane waves propagating through a linear medium, the superposition principle applies. Consider D plane waves $f_k(t;\mathbf{d})$ with propagation vectors given by $\mathbf{v}(\theta_k,\phi_k), k = 1,\cdots,D$. With additive noise $\mathbf{n}(t) \in \mathbb{C}^{M\times 1}$ at the M sensors, the signal model given by (1.2) becomes

$$\mathbf{y}(t) = \sum_{k=1}^{D} \begin{bmatrix} f_k(t-\tau_1^{(k)}) \\ f_k(t-\tau_2^{(k)}) \\ \vdots \\ f_k(t-\tau_M^{(k)}) \end{bmatrix} + \mathbf{n}(t) \tag{1.4}$$

and the narrowband model (1.3) becomes

$$\mathbf{y}(t) = \mathbf{A}(\Theta,\Phi)\mathbf{s}(t) + \mathbf{n}(t) \tag{1.5}$$

where $\mathbf{A}(\Theta,\Phi) \in \mathbb{C}^{M\times D}$ is known as the **Array Manifold** matrix whose columns are comprised of D steering vectors $\mathbf{a}(\theta_k,\phi_k), k = 1,\cdots,D$, and the vector $\mathbf{s}(t) \in \mathbb{C}^{D\times 1}$ contains the D complex envelopes. For the special case of $1D$ arrays, the position vectors of the elements are given by $\mathbf{d}_m = [0 \quad 0 \quad d_m\lambda/2]^T, m = 1,2,\cdots,M$ where λ is the carrier wavelength. In this case, the steering

vector corresponding to the direction (θ, ϕ) depends only on the elevation angle θ and is given by

$$\mathbf{a}(\theta) = \begin{bmatrix} j\pi d_1 \sin\theta \\ e^{j\pi d_2 \sin\theta} \\ \vdots \\ e^{j\pi d_M \sin\theta} \end{bmatrix}$$

1.1.1.2 Autocorrelation and Spectral Estimation

The spatio-temporal signal model received by an array of sensors often has a statistical representation. For the narrowband signal model, this implies that the waveforms $\{s_k(t)\}_{k=1}^{D}$ are represented as jointly Wide Sense Stationary (WSS) random processes with a correlation matrix

$$\mathbf{R}_{ss}(\tau) \triangleq \mathrm{E}[\mathbf{s}(t)\mathbf{s}(t-\tau)] \tag{1.6}$$

For narrowband signal model, the signals are primarily temporal stochastic processes. However, for wideband signals, the model corresponds to a random field [96] and the autocorrelation matrix of WSS $\mathbf{s}(t;\mathbf{d})$ has a spatio-temporal structure $\mathbf{R}_{ss}(\tau,\boldsymbol{\delta}) = \mathrm{E}[\mathbf{s}(t;\mathbf{d})\mathbf{s}(t-\tau;\mathbf{d}-\boldsymbol{\delta})]$. Under the narrowband WSS assumption, the autocorrelation matrix is given by

$$\mathbf{R}_{yy}(\tau) = \mathbf{A}(\Theta,\Phi)\mathbf{R}_{ss}(\tau)\mathbf{A}^H(\Theta,\Phi) + \sigma_n^2\mathbf{I}$$

where the noise is assumed zero mean, spatially white (uncorrelated across sensors) with power σ_n^2 and it is uncorrelated with the signals. The correlation matrix for $\tau = 0$ is simply denoted $\mathbf{R}_{yy}$. This correlation matrix plays a fundamental role in several different types of estimation techniques used in array processing. **Spectral Estimation** broadly refers to the family of estimation techniques that explicitly uses this correlation matrix. In the traditional sense, spectral estimation implies deriving information from the spectrum of a signal by computing the Fourier Transform of its autocorrelation sequence. Spectral analysis finds application in speech processing for speech analysis and representation, in source localization and tracking for radar and sonar systems, for predicting periodicites in data used in economics, seismology, meteorology etc. However, the family of spectral estimation algorithms in array processing extends beyond the conventional Fourier based interpretation of the spectrum and involves computing spectrum-like functions of parameters of interest

(also called pseudo-spectrum) from the autocorrelation function.

1.1.2 Parametric Estimation

The random multichannel signal model $\mathbf{y}(t)$ carries both *spatial and temporal information* in the form of unknown quantities of interest such as the number of sources D, the Directions of Arrival (DOA) $\{(\Theta_k, \Phi_k)\}_{k=1}^{D}$, the signal waveforms $s_k(t)$, and the noise power σ_n^2. A central goal in array processing is to develop powerful algorithms to estimate or extract information from the spatio-temporal observation.

The parametric approach to estimation assumes a probabilistic model for the underlying signal and applies Maximum Likelihood (ML) estimation to extract the unknown parameters of the distribution. Two different assumptions about the source signals $\mathbf{s}(t)$ lead to different ML approaches. They are known as deterministic ML (DML) and stochastic ML (SML) algorithms. For the deterministic model, the source signals are modeled as unknown deterministic quantities and the randomness of $\mathbf{y}(t)$ in (1.3) is entirely due to the noise with unknown variance σ_n^2. Modeling the noise as spatially white circularly symmetric Gaussian random vector, one can derive the Deterministic ML (DML) estimate [195] $(\hat{\Theta}, \hat{\Phi}, \hat{\sigma_n^2}, \{\mathbf{s}(t)\}_{t=1}^{L})$ as one that maximizes the log-likelihood function computed over L time samples

$$-L \log \sigma_n^2 + \frac{1}{\sigma_n^2 L} \sum_{t=1}^{L} \|\mathbf{y}(t) - \mathbf{A}(\Theta, \Phi)\mathbf{s}(t)\|^2$$

The stochastic approach however consists of modeling the sources as jointly WSS Gaussian random processes with covariance matrix $\mathbf{R}_{ss}$. The negative log likelihood function is shown proportional to [17]

$$\frac{1}{L} \sum_{t=1}^{L} \|\mathbf{\Pi}^{\perp}_{\mathbf{A}(\Theta,\Phi)}\mathbf{y}(t)\|^2$$

where the notation $\mathbf{\Pi}^{\perp}_{\mathbf{P}}$ denotes the projection matrix onto the orthogonal complement of the range space of the matrix $\mathbf{P}$. The SML algorithm exhibits better large sample accuracy than corresponding DML algorithms [168]. The SML algorithm also attains the Stochastic Cramer-Rao lower bound (CRB) [169] for Gaussian signals whereas the DML cannot consistently estimate the source parameters and they do not attain the deterministic CRB. Although the ML estimates exhibit higher ac-

curacy, robustness and optimality (in terms of achieving CRB), they are computationally expensive and require multidimensional search over the parameter space. In contrast, the spectral methods as described below provide a computationally attractive and simpler alternative to the ML methods.

1.1.3 Spectral Estimation

As mentioned earlier, the spectral approaches largely operate on the spatial correlation matrix $\mathbf{R}_{yy}$ and compute spectrum-like functions for estimation of the parameters of interest. They can be broadly classified as (a) beamforming, and (b) subspace-based methods. The former is linear while the latter is non linear and has distinct advantages over beamforming.

Beamforming [187] is a spatial filtering technique where the observed signals at the output of M sensors are linearly weighted by a weight vector $\mathbf{w} \in \mathbb{C}^{M\times 1}$ and then summed to produce a scalar output. In its simplest form, the weight vector $\mathbf{w}$ is chosen to steer the array to a particular direction (θ_0, ϕ_0). This is obtained by choosing $\mathbf{w} = \mathbf{a}(\theta_0, \phi_0)$ and computing the output as

$$z(t) = \mathbf{w}^H \mathbf{y}(t).$$

By changing $\mathbf{w}$ to steer at different directions, one obtains an estimate of the spatial spectrum as

$$P(\theta, \phi) = \mathbf{a}^H(\theta, \phi)\Big(\sum_{t=1}^{L} \mathbf{x}(t)\mathbf{x}^H(t)\Big)\mathbf{a}(\theta, \phi)$$

A different type of beamforming based on an optimal weighting was proposed by Capon [30, 209] and it is called the minimum variance distortionless beamformer (MVDR) which solves for $\mathbf{w}$ as

$$\min_{\mathbf{w}} P(\mathbf{w})$$
$$\text{such that } \mathbf{w}^H \mathbf{a}(\theta_0, \phi_0) = 1$$

where $P(\mathbf{w}) = \mathbf{w}^H\Big(\sum_{t=1}^{L} \mathbf{x}(t)\mathbf{x}^H(t)\Big)\mathbf{w}$. This optimization tries to minimize the power from noise and any signal coming from direction other than (θ_0, ϕ_0). The direction (θ_0, ϕ_0) is also known as the "look direction". MVDR beamforming has better resolution and noise suppression capabilities than the steering approach. By measuring the power at the output of the beamformer designed for a certain "look direction", once can infer the presence of a potential source and estimate its direction of arrival (DOA). For an extensive discussion on various other beamforming techniques,

including the adaptive versions, the interested reader is referred to [187].

In contrast to beamforming, subspace based methods for spectral estimation rely upon the eigen structure of the correlation matrix to estimate the spatial paremeters (DOA) of interest. Pisarenko's harmonic retrieval [152] was one of the earliest algorithms belonging to this family. However, the MUSIC algorithm [160] is probably the most popular subspace based technique for direction finding which gave rise to a family of other algorithms exploiting the low rank structure of the covariance matrix. The covariance matrix $\mathbf{R}_{yy} = \mathbf{A}\mathbf{R}_{ss}\mathbf{A}^H + \sigma_n^2\mathbf{I}$ (we drop the dependency of $\mathbf{A}$ on Θ and Φ) can be decomposed (via eigenvalue decomposition) into $\mathbf{U}_s\mathbf{\Lambda}\mathbf{U}_s^H + \sigma_n^2\mathbf{U}_n\mathbf{U}_n^H$. The columns of $\mathbf{U}_n \in \mathbb{C}^{M\times(M-D)}$ are orthogonal to the range space of $\mathbf{A}$, denoted $\mathcal{R}(\mathbf{A})$. The MUSIC spatial spectrum is defined as

$$P_M(\theta,\phi) = \frac{1}{\mathbf{a}^H(\theta,\phi)\mathbf{\Pi}_n\mathbf{a}(\theta,\phi)}$$

where $\mathbf{\Pi}_n$ denotes the projection operator on $\mathcal{R}(\mathbf{U}_n)$. It can be shown that for certain array geometries, the MUSIC spectrum exhibits peaks exactly at the true DOA locations with no false or missing peaks. Note that $P_M(\theta,\phi)$ is not a true spectrum but represents a measure of distance between two subspaces. Also notice that estimating the DOAs now merely involves a $2D$ search over (θ,ϕ) space. There are several other families of subspace based methods such as ESPRIT, Min-Norm algorithm, Root-MUSIC, IQML etc. One important observation on subspace based methods is that the number of sources D need to satisfy $D < M$. Beamforming algorithms also have this assumption implicitly. Hence, *most estimation algorithms in array processing consider an overdetermined signal model (1.3) where* $M > D$. This indicates that the number of sources D that can be resolved is upper bounded by the number of sensors M.

1.1.4 Sampling and Underdetermined Estimation

While a large body of literature in array processing only considers the overdetermined signal model, a few lines of work consider the underdetermined model where the number of unknown sources can be actually less than the number of physical sensors. In all these cases, *the geometry of the array or the spatial samples plays a vital role in efficient representation of the underdetermined system*. Most of these techniques are based on creating the effect of a *virtual array* with many more elements than the physical array by using suitable sampling strategies. The virtual array, also known as a co-

array, can be either a sum co-array for an active sensing system or a difference co-array for a passive sensing system. The number of elements in the co-array plays the primary role in determining how many sources can be resolved. For an array with M physical sensors, both the sum and difference co-array can have upto $O(M^2)$ elements, depending on the geometry of the underlying physical array.

One popular application which explicitly uses a virtual array is that of synthetic aperture radar (SAR) where the effect of an artificial larger aperture is created by virtue of the motion of the radar [165]. The use of sum and difference co-arrays respectively in coherent and incoherent imaging was extensively discussed in [89]. The difference co-array also arises in direction finding problems in array processing when one considers the correlation between the signals received at the output of the antennas. Underdetermined models for direction finding using only the second order statistics (correlation) was proposed in [149, 150] and thoroughly investigated in [1, 2]. They propose the use of special one dimensional array geometry known as the minimum redundancy array (MRA) [126] (we shall review this structure in Chapter 2). Given a fixed number (say, M) of sensors, MRA is theoretically the optimum linear geometry which produces the maximum possible number of elements in the difference co-array *without any missing elements in between.* However, there is no closed form expression for the structure of the MRA or the number of elements in its difference co-array and hence this geometry is not analytically tractable. Also, there is no obvious way to extend this geometry to higher dimensions, or to the so-called higher order difference co-arrays (which arise in estimation problems utilizing higher order statistics of the signal). In the latter case, a number of different virtual arrays can arise depending on the order of statistics used. They constitute the general class of *higher order difference co-arrays* [42, 43]. In the context of active sensing, MIMO radar [38, 39, 16] explicitly utilizes the concept of a sum co-array to create the effect of a virtual array by transmitting orthogonal waveforms and subsequently processing them with a bank of matched filters at the receiver.

1.1.5 New Questions in Array processing

The preceding discussion on existing approaches to underdetermined estimation in array processing brings us to the following questions.

1. **Is there a simple systematic geometry for underdetermined array processing?** In particular, we would want to develop an analytically tractable array geometry (as opposed to the MRA

which has no closed form) that has a simple structure and is capable of producing $O(M^2)$ elements in its difference co-array. Also, it is desirable that the structure can be generalized to higher dimensions and be easily extended to the case of higher order statistics. In this thesis, we will propose such a geometry, namely the nested arrays, which will be shown to satisfy all these conditions.

2. **Is there an algorithm that can exploit the co-array?** Previous attempts at this problem have used variations of an augmented covariance matrix [1, 2, 149], which, for finite number of measurements, is unfortunately not positive semidefinite and hence does not represent a valid covariance matrix. In this thesis, we will propose a new subspace based algorithm which will produce a positive semidefinite matrix representing the covariance matrix corresponding to the co-array even for finite number of snapshots. This algorithm will be shown to be generic enough to be applied to two dimensional arrays, as well as to the case of higher order statistics.

3. **Can the ideas in one dimension be extended to multiple dimensions?** Array processing in two dimensions ($2D$) can be considerably different from its one dimensional ($1D$) counterpart. Also, there can be numerous geometries for spatial sampling in $2D$, which do not arise in $1D$. In this thesis, we will show that the proposed nested array can be easily extended to $2D$ if we consider sampling on $2D$ lattices. We will also consider another sampling strategy, namely, coprime sampling, which when extended to $2D$, necessitates the use of coprime and commuting integer matrix pairs. This gives rise to theoretical questions on the existence and construction of such matrix pairs which will be extensively discussed in Chapter 6.

4. **How many sources can be resolved using higher order statistics?** It is well known that by exploiting higher order statistics, it is possible to identify an increasingly large number of sources, which far exceeds the number of sensors. Previously known results indicate that higher order co-arrays play an important role in this regard and the number of identifiable sources depends on different possible configurations of the co-array. In particular, it was known that by using $2q$th order statistics ($q > 1$ is an integer) and M physical sensors, it is possible to resolve upto $O(M^q)$ sources. In this thesis, we will show that fundamentally, it is possible to overcome this limit and identify upto $O(M^{2q})$ sources by using an extension of the nested structure.

1.2 Sparse Estimation and Compressive Sensing

The problem of underdetermined estimation has received great attention in recent times particularly due to its connection with the theory of sparse sampling and recovery, also popularly known as "Compressive Sensing". Fundamentally, it seeks to solve for $\mathbf{x} \in \mathbb{C}^{N\times 1}$ from

$$\mathbf{y} = \mathbf{A}\mathbf{x} \tag{1.7}$$

where $\mathbf{A} \in \mathbb{C}^{M\times N}$ is a fat matrix with $N \gg M$. As such, the problem is ill defined and the number of solutions is infinite. However, with more assumptions on the particular $\mathbf{x}$ that is desired, the problem actually can permit a unique answer. A very common assumption is that of *sparsity*.

1.2.1 The Ubiquitous Sparse Model

Sparsity is a fundamental attribute which characterizes many signals: natural and man-made. A signal is said to be sparse if it only has a few (almost) non zero coefficients. The sparsity in signals can be visible either in the observed domain, or in a transformed domain. The particular transformation under which a signal exhibits sparsity often depends on the specific application of interest. A wide variety of signals and measurements can have a natural sparse representation. Examples include images and videos, spatio temporal spectrum of data collected by radar and sonar systems, anomalies and outliers in data used in statistical regression and inference, fault and outages in networked systems etc. Knowledge of the sparsity of the signal has been exploited to perform (sub-Nyquist) sampling and dimension reduction of data, which allows one to represent the sparse signal in a compact fashion. This in turn implies lower computational and storage requirements which are particularly attractive in many applications of today which require storage and processing of massive amounts of data [8]. The area of "Compressed Sensing" [56, 24] has emerged to address the interconnections of sparsity, sampling, and reconstruction and provides an alternate view to traditional approaches to sampling. There are mainly two aspects of this area: the first is that of acquiring fewer measurements and performing dimension reduction of data with the knowledge that the underlying signal is sparse. The second is to reconstruct the original signal from the small number of (compressed) measurements that are collected. The second problem is also termed as the sparse recovery problem and has a direct connection to the underdetermined estimation problem (1.7). It arises in a large number of applications such as sub Nyquist sampling

of multi band signals [125], image and video processing [117, 67], radar [87, 40], multivariate regression [130], sparse Bayesian estimation [95], monitoring and inferring from networked data [84], and channel estimation [48, 7].

1.2.2 A Brief Review of Sparse Estimation

Sparse estimation is concerned with finding the sparsest $\mathbf{x}$ satisfying (1.7). The problem can be readily solved if we knew which elements in $\mathbf{x}$ are non zero. However, lack of this knowledge makes the problem very difficult to solve in a computationally efficient manner. The problem of recovering the sparsest solution to (1.7) is also known as the l_0 minimization problem and is often cast as $\min_{\mathbf{x}:\mathbf{y}=\mathbf{A}\mathbf{x}} \|\mathbf{x}\|_0$ where $\|\mathbf{x}\|_0$ denotes the number of non zero elements in $\mathbf{x}$. This problem in its most general form is NP hard. An alternative approach is to solve the l_1 norm minimization problem: $\min_{\mathbf{x}:\mathbf{y}=\mathbf{A}\mathbf{x}} \|\mathbf{x}\|_1$ and hope the solution is the same as that obtained from the l_0 problem. The l_1 norm $\|\mathbf{x}\|_1$ denotes the sum of absolute values of $\mathbf{x}$ and it is the smallest p such that the $p-$norm $\|\mathbf{x}\|_p$ is convex. The l_1 minimization problem is a linear program which can be solved efficiently by algorithms with polynomial time complexity. In fact, l_1 minimization has been used for several decades in order to seek sparse solution to the underdetermined problems. However, for a long time, there was a lack of rigorous analytic results for equivalence of the l_1 and l_0 problems. These results were finally established in the 2000s which led to the emergence of "Compressed Sensing" [24, 56] as an active field of research. It has been shown that under certain conditions on the matrix $\mathbf{A}$, the solution obtained from l_1 minimization problem is identical to that from the l_0 norm minimization problem. These conditions are characterized by different properties of the matrix $\mathbf{A}$ such as mutual coherence [58], Restricted Isometry Property (RIP) [25], Exact Reconstruction Condition (ERC) [177] and the Null Space Property [47]. In the following, we present a brief history of algorithmic developments for sparse estimation over the last few decades.

1.2.2.1 A Brief History of Sparse Recovery Algorithms

Sparsity promoting l_1 regularized techniques were commonly used for deconvolution in marine seismology in the 1970's [32, 173]. In the 1990's, the Basis Pursuit (BP) [34] algorithm was proposed by Chen and Donoho for l_1 minimization. The goal was to perform atomic decomposition of signals in overcomplete dictionaries. Independently, the FOCUSS algorithm was proposed [83] as a re-weighted sparsity-enforcing norm minimization approach with applications in neuromag-

netic imaging. As an alternative to the convex l_1 minimization, subspace based techniques such as MUSIC [160] were also introduced for recovery of multi band signals with sparse spectrum [76]. Meanwhile, Matching Pursuit (MP) [122, 52] was introduced as a greedy algorithm for denoising signals in redundant dictionaries. Tibshirani proposed a l_1 regularized least squares minimization LASSO [175] for sparse regression in a noisy setting. In what was to follow, extensions of MP and LASSO would become popular means for sparse recovery with analytically provable guarantees.

It was not until the 2000s that a series of breakthrough results demonstrated that the NP hard l_0 minimization problem can be exactly solved by the convex l_1 minimization problem, under certain conditions on the matrix $\mathbf{A}$. As a result, the field of Compressive Sampling [56, 24] emerged, partly building upon the past work (mostly empirical) on sparse recovery, and largely due to the analytic results on the equivalence of l_0 and l_1 minimization. These results are undoubtedly among the most important contributions in recent times. In order to establish the power of l_1 minimization in recovering the sparse solution, theoretical guarantees based on mutual coherence of $\mathbf{A}$ were developed in [59, 60, 58] proving the stability of the BP algorithm in presence of noise. Independently, a series of theoretical results on l_1 minimization [28, 27, 26] provided sufficient conditions for exact reconstruction of sparse vectors using random measurement matrices, in terms of the newly coined term "Restricted Isometry Property" (RIP) of the matrix $\mathbf{A}$ [28, 25]. In particular, it was shown that exact reconstruction of s-sparse vectors is possible using random matrices as projection operators provided the size of projections is $O(s \log N)$. Such analysis was also extended for the noisy scenario [27, 26]. The work of Donoho et al established necessary and sufficient conditions for sparse recovery and provided stronger guarantees than RIP by using the notion of *k-neighborly* polytopes [57, 61, 62]. For noisy observations, the performance of LASSO was well studied in [190, 205]. A set of necessary and sufficient conditions were established for the solution of LASSO to yield the true support, either exactly or with high probability. A number of papers have also focused on deriving information theoretic bounds on the possibility of recovering sparse vectors, irrespective of any particular algorithm used. These formulations consider a statistical model and derive probabilistic guarantees on the sparse recovery by deriving fundamental relations between number of measurements, size of dictionary, sparsity and the noise level. The results of [189, 5, 172] develop such guarantees.

Alongside the convex formulation (l_1 minimization, LASSO etc) of the sparse recovery problem, a family of greedy algorithms has also been revisited and analyzed to yield guarantees which are

comparable to the convex counterparts. Most notable among these is the Matching Pursuit (MP) algorithm [122]. It was modified to Othogonal Matching pursuit (OMP) [147, 52] in the 1990s to yield superior performance. OMP was revisited by Tropp [177, 180] in the context of sparse recovery and thoroughly analyzed. A sufficient condition called the Exact Reconstruction Condition (ERC) was proposed under which OMP can exactly solve the l_0 problem. It was also shown that for random Gaussian ensembles, the number of measurements required by OMP to recover sparse vectors with high probability, is the same as BP.

1.2.2.2 Extensions

While most algorithms cast the sparse recovery problem in a deterministic setting, a few of them considers a Bayesian approach for the sparse estimation problem. The work in [176, 95, 199, 207] belongs to this class of algorithms. They use a sparse Bayesian learning approach to formulate a MAP estimate for the sparse unknown vector. The sparsity is explicitly characterized in terms of a weight function (which can be zero) on the distribution of $\mathbf{x}$ and it is considered as a hyperparameter in the estimation problem which can be "learnt" via various approaches. The learning algorithms can be shown to enforce sparsity and recover the true solution. This body of work constitutes an alternative to the majority of existing work on convex relaxation based sparse recovery techniques.

Another extension of the sparse recovery problem deals with Multiple Measurement Vectors (MMV) as opposed to the SMV model given by (1.7) where a set of multiple unknown sparse vectors shares a common sparse support. Some of the algorithms developed for SMV can be extended to this case in a more or less straightforward fashion. These examples include M-FOCUSS [48], M-OMP [181, 128], mixed norm convex optimization, [121, 36] and M-SBL [200]. However, in the MMV formulation it is possible to exploit additional attributes provided by the multiple vectors such as rank [53], correlation [207] etc. Considerable work has been done to extend the theoretical analysis for SMV to the MMV problem. Fundamental guarantees on recovery from a uniqueness perspective were established in [36]. A more probabilistic argument for sparse recovery in MMV was made in [71] where the sparse signal was assumed to have a certain probability distribution and an average case analysis was performed. The extension of LASSO to MMV, known as multivariate group LASSO, was analyzed in [130] and necessary and sufficient conditions were derived for support recovery in the noisy setting. The work in [172] established sufficient conditions for

sparse recovery by using multiple hypothesis testing framework and also derived a converse result by using information theoretic argument.

1.2.3 New Questions in Sparse Estimation

In this thesis, we will address some new questions in sparse estimation from a statistical point of view. Bayesian approaches to compressive sensing [95, 176] provide one way of incorporating a statistical model and form a fundamental connection between statistical estimation and sparse reconstruction theory. This perspective gives rise to many new questions that can be asked beyond the purview of existing sparse sampling and recovery techniques. In this thesis, we will specifically ask the following questions and explore new possibilities along their lines:

1. **What kind of priors can we exploit besides sparsity?** The solution to underdetermined systems becomes unique under the prior knowledge about sparsity of the solution. In the literature, several other forms of prior have been considered such as non uniform support [99], partial knowledge of support [188], model based approaches [9] or use of rank of the data [53]. However, the statistical framework offers the possibility of using a much wider variety of priors. In this thesis, we will consider such a prior in terms of *correlation between the variables* and develop new theoretical guarantees.

2. **How can we reduce the requirement on the number of samples?** The answer to this question will constitute a major contribution of this thesis. Almost every approach to sparse recovery, deterministic or Bayesian, seems to indicate that the number of samples M cannot be less than the sparsity level to be recovered. But, we will show that by designing suitable sampling schemes, it is possible to recovery sparsity levels as large as $O(M^2)$ by exploiting the statistical priors.

3. **How can Sampling and Reconstruction Exploit Priors?** In the underdetermined model (1.7), knowledge of statistical priors related to the probability distribution $p(\mathbf{x})$ can lead to new sampling matrix $\mathbf{A}$ as well as recovery algorithms that are capable of exploiting such knowledge. Common sampling matrices used in compressive sensing are mostly drawn from random ensembles. However, depending on the application, they can often exhibit more structure. In this thesis, we will also show how sparse sampling schemes can fundamentally utilize knowledge of priors and develop new reconstruction algorithms.

1.3 Outline and Scope of the Thesis

In this dissertation, we visit the relatively less explored frontiers of underdetermined estimation theory in the context of statistical signal processing as well as sparse reconstruction theory. There are two main problems considered in this thesis. The first problem of focus is estimation in antenna array processing (Chapters 2, 3, 4, 5 and 6). We consider a strictly underdetermined signal model where the number of sensors or antennas can be considerably fewer than the number of unknown spatial parameters. Different variations of this problem are studied. They include correlation based direction finding and extension to distributions with higher order statistics, beamforming and direction finding for wideband signals, and direction finding for two dimensional arrays. We propose novel spatial sampling structures in one and multiple dimensions and develop new algorithms to solve the underdetermined estimation problem in an efficient and systematic fashion. The second part of the thesis considers the problem of sparse recovery in compressive sensing (Chapters 7 and 8). By introducing a statistical framework to this problem, we will develop new results for the problem of sparse sampling and reconstruction which will be shown to have better performance guarantees than existing ones. Below is a brief description of the scope of each chapter.

1.3.1 Nested Arrays in One Dimension (Chapter 2)

While most estimation and detection algorithms in array processing consider an overdetermined observation model, the focus of this thesis is the underdetermined signal model. In Chapter 2, we consider the problem of direction finding using such a model where the number of antennas can be substantially fewer than the number of unknown spatial parameters (angles of arrival). For unique identification of parameters in this model, we propose the *use of correlation priors* and derive a model based on the correlation of the observations which retains all the information about the parameters but offers a more explicit representation which is amenable to underdetermined estimation. The traditionally used Uniform Linear Array (ULA) which is analogous to uniform sampling, is shown to be an unsuitable choice for the purpose of identifying more sources than physical sensors. A new sparse array geometry, namely, the nested array is proposed [135] which is shown to be fundamentally capable of resolving more sources than the number of physical sensors by exploiting the so called "difference co-array" that arises in the correlation based framework. The geometry is shown to be simple and systematic and easily extendable to more than one dimension. An optimal array

structure is derived for maximizing the degrees of freedom offered by such an array. It is shown to be order optimal over any array geometry in terms of the maximum number of resolvable sources using only the correlation of the data. We also develop a new subspace based algorithm which is capable of estimating the directions of arrival from the covariance matrix of the nested array. Further applications of the nested array for beamforming in the power domain are also demonstrated. The nested array thereby provides a new sampling approach to array processing with more sources than sensors.

1.3.2 Nested Arrays in Two Dimensions (Chapter 3)

In Chapter 3, we extend the $1D$ nested array to two dimensions for two dimensional ($2D$) direction finding (elevation and azimuth) problems [142, 143]. $2D$ array processing differs considerably from that in one dimension due to some unique problems that arise only in two dimensions. Among many possible sampling geometries in $2D$, we focus on lattices and propose a family of two dimensional Nested Arrays on Lattices. The concept of $1D$ nested array is extended to $2D$ lattices via integer matrices. Various orientations of the proposed structure are considered and optimized. It is further shown that the construction of nested arrays for separable and non separable lattices can be directly related via Smith Form of integer matrices. We also develop a new subspace algorithm for $2D$ direction estimation using the nested structure. One of the unique issues in two dimensions is the absence of exact conditions on parameter identifiability. We address this problem and develop new insights into two dimensional parameter identification and prove guarantees for different scenarios.

1.3.3 Higher Order Statistics and Extension of Nested Arrays (Chapter 4)

In Chapter 4, we consider the problem of underdetermined estimation using higher order statistics (HOS) of the signal. Higher order statistics have played important roles in statistical estimation from an underdetermined model since they can provably allow estimation of more parameters than the number of physical measurements. We revisit this problem in the context of direction finding in antenna arrays and demonstrate that it is possible to further improve upon the performance of existing methods utilizing HOS [140]. This is shown to be possible by identifying a new virtual array structure embedded within the higher order statistics estimated from such an underdetermined model. This virtual array is a generalization of the difference co-array and its size is shown to be

proportional to the order of the statistics used. However, the size also depends on the underlying physical array and the ULA is again shown to be a sub-optimal choice. We propose an extension of the nested array which works for every order of statistics and the size of its virtual array is shown to be order optimal. This demonstrates the flexibility of the proposed nested array in being adapted to every order of statistic. The subspace algorithm proposed in Chapter 2 is also shown to be extensible to this case and it can provably resolve larger number of sources than the previously known algorithms. We also revisit Cramer Rao Bound analysis for underdetermined estimation and examine its applicability to the current model under both deterministic and stochastic setting. The performance of the extended nested array coupled with the new subspace algorithm is shown to be significantly better than previously known approaches, particularly when large number of temporal samples are available.

1.3.4 Wideband Beamforming and Estimation (Chapter 5)

The problems of direction finding and beamforming for wideband signals are significantly different from their narrowband versions. This is due to the fact that the response of the array is different at every frequency and incorporation of the information from all different frequencies in the most effective way remains a challenging task. In Chapter 5, we propose two new approaches for wideband array processing. One of them focuses on direction finding from wideband signals [131, 132]. The other one explores new directions in frequency invariant beamforming where the goal is to make the beampattern of the array appear identical across the frequencies. Our approach to wideband direction finding is based on coherently combining the signals in the subbands before performing the estimation. Usually such combination requires some coarse estimtates of the directions. However, we propose a new approach to coherent combination DOA estimation which does not require such estimates. This approach, called autofocusing, is shown to provide better performance when the coarse estimates are not available. In the second part, we develop new approaches to *frequency invariant* beamforming using $2D$ lattices for wideband signals. Wideband beamforming is usually realized as a spatio-temporal filtering process. However, we show that such beamforming can be made fully spatial by using two dimensional arrays and striking a trade off between time and space. We further show that it is possible to realize the $2D$ arrays virtually by using far fewer number of suitably placed physical sensors. This shows that frequency invariant beamforming can indeed be performed without temporal processing or deployment of more sensors.

1.3.5 Commuting Coprime Matrix Pairs (Chapter 6)

Coprime sampling [184, 185] has been recently proposed as a sub Nyquist sampling scheme which enables the autocorrelation sequence to be sampled at the Nyquist rate. Hence, one can employ significantly slower A/D converters (and save power) or deploy fewer sensors and yet retain all the information about the spectrum of the signal. Coprime sampling when extended to two dimensions, requires the use of integer matrix pairs which are commuting and coprime. It is challenging to fully characterize the entire family of such matrices. In this chapter, we investigate several families of commuting matrix pairs and establish necessary and sufficient conditions for their coprimality [137]. In particular, we look into circulant, skew circulant, adjugate and triangular pairs of matrices and derive simplified conditions on their entries. The results obtained in this chapter can lead to constructive techniques for commutive coprime matrices.

1.3.6 Correlation Aware Sparse Recovery (Chapters 7 and 8)

In Chapters 7 and 8, we study the problem of underdetermined sparse recovery by bringing in a statistical model. Most approaches to sparse recovery have been deterministic in the sense that they make no assumptions on the statistics of the unknown $\mathbf{x}$ in (1.7). The Bayesian approaches however [176, 95] assume a multivariate normal distribution on $\mathbf{x}$ but do not consider priors other than sparsity. We consider a generic model for sparse recovery which includes a variety of applications including direction finding for antenna arrays. We introduce a framework where we consider priors on the unknown sparse signal in the form of statistical correlation [141]. Given this prior, a model based on the ideal correlation matrix is developed in Chapter 7 for recovering the sparse support. Fundamental conditions on the uniqueness of recovery are explored which give rise to novel conditions on the measurement matrix and lead to new design challenges. We demonstrate that the proposed nested and coprime arrays prove to be a natural candidate for sparse recovery with correlation priors. We also derive necessary and sufficient conditions on the relaxed problem by making a connection to Donoho's result on neighborly polytopes [62]. In Chapter 8, we consider an extension of this framework in the form of a Multiple Measurement Vector (MMV) model. We use this model to analyze the performance of the proposed correlation aware sparse recovery technique when we only have estimates of the correlation computed from the MMV [144]. We demonstrate that in this case it is possible to have a probabilistic guarantee for support recovery. For Gaussian sources, it is shown that such recovery can happen with overwhelming probability with respect

to the number of measurement vectors available [145]. This chapter provides a first step towards explicit use of correlation priors in sparse recovery and leads to many future directions for prior aware sparse recovery techniques.

1.4 Notations

Matrices are denoted by uppercase letters in boldface (e.g., $\mathbf{A}$). Vectors are denoted by lowercase letters in boldface (e.g., $\mathbf{a}$). Superscript ^{H} denotes transpose conjugate, whereas superscript * denotes conjugation without transpose. The notation $[\mathbf{A}]_{i,j}$ denotes the (i,j)th elements of matrix $\mathbf{A}$. For a set S, $\mid S \mid$ denotes its cardinality. Similarly, given two sets S_1 and S_2, $S_1 \setminus S_2$ denotes the difference of the sets, $S_1 \cap S_2$ denotes their intersection and $S \cup S_2$ denotes their union. Given a set of integers $S \subset \{1, 2, \cdots, N\}$ and a vector $\mathbf{v} \in \mathbb{C}^{N \times 1}$, the vector $[\mathbf{v}]_S \in \mathbb{C}^{|S|} \times 1$ consists of elements of $\mathbf{v}$ indexed by S. Similarly given a matrix $\mathbf{A} \in \mathbb{C}^{M \times N}$, the matrix $\mathbf{A}_S \in \mathbb{C}^{M \times |S|}$ consists of the columns of $\mathbf{A}$ indexed by S.

The symbol $\odot$ denotes the Khatri-Rao product [186] between two matrices of appropriate size and the symbol $\otimes$ is used to denote the left Kronecker product (p. 1353 of [186]). For two matrices $\mathbf{A}$ and $\mathbf{B}$ of same dimensions, $\mathbf{A} \circ \mathbf{B}$ denotes the Hadamard product [91]. For example, consider 2×2 matrices $\mathbf{A} \in \mathbb{C}^{M \times N} = \begin{bmatrix} a_1 & a_2 \\ a_3 & a_4 \end{bmatrix}$ and $\mathbf{B} \in \mathbb{C}^{M \times N} = \begin{bmatrix} b_1 & b_2 \\ b_3 & b_4 \end{bmatrix}$. Then, the Kronecker, Khatri Rao, and Hadamard products are respectively given by

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} \mathbf{A}b_1 & \mathbf{A}b_2 \\ \mathbf{A}b_3 & \mathbf{A}b_4 \end{bmatrix}, \quad \mathbf{A} \odot \mathbf{B} = \begin{bmatrix} a_1 b_1 & a_2 b_2 \\ a_1 b_3 & a_2 b_4 \\ a_3 b_1 & a_4 b_2 \\ a_3 b_3 & a_4 b_4 \end{bmatrix}, \quad \mathbf{A} \circ \mathbf{B} = \begin{bmatrix} a_1 b_1 & a_2 b_2 \\ a_3 b_3 & a_4 b_4 \end{bmatrix}$$

Chapter 2

Nested Arrays in One Dimension

2.1 Introduction

In this chapter, we will consider a classical parameter estimation problem, namely simultaneous direction finding using antenna arrays. Contrary to the traditional scenario, we will consider the case where the model is highly underdetermined. This happens when the number of spatial samples of the impinging electromagnetic waveform (equivalently the number of antennas) is less than the number of unknown directions to be resolved. We will develop a systematic theory of such an estimation problem and propose a new antenna geometry which is fundamentally capable of resolving more sources than the number of antennas. This underdetermined model will be revisited in Chapter 7 in the context of finding sparse solutions for linear underdetermined system of equations. This problem arises in a wide variety of applications such as sub Nyquist sampling and reconstruction, multivariate regression, imaging, source localization etc. The proposed spatial nested sampling will be shown to play an important role in such problems as well.

Antenna arrays perform spatial sampling of impinging electromagnetic waves and the samples are used to perform a wide variety of tasks including detection and estimation of spatial signatures. Direction-of-arrival (DOA) estimation and beamforming constitute two major problems of interest in this regard. However, traditionally, both of these problems are formulated based upon a signal model that uses a uniform linear array (ULA) and considers a scenario where the number of sources is less than the number of sensors. The main reason behind this is that the maximum number of sources identifiable with a N element ULA using traditional subspace based methods like MUSIC [160] is $N-1$. In this chapter, we explore the class of non uniform arrays and propose a novel array structure which is capable of providing a dramatic increase in the degrees of freedom (DOF) and

hence can resolve significantly more sources than the actual number of physical sensors. We call this class of arrays as "nested arrays" because they are obtained by combining two or more ULAs with increasing inter-sensor spacing. We shall demonstrate that using only second order statistics of the impinging sources, it is possible to obtain $O(N^2)$ DOF from only N physical sensors.

The main idea behind the ability to resolve more sources than physical sensors is based upon the concept of a *difference co-array*. This helps in performing array processing with increased degrees of freedom in a completely passive scenario. The difference co-array becomes visible as one computes the correlation between the signals received at the different antennas. Hence the new direction finding and beamforming algorithms proposed in this chapter work entirely in the correlation domain. The co-array concept has previously been treated for specific array geometries in [89],[103]. In Section 2.5, we propose our nested array structure which can greatly increase the degrees of freedom of the corresponding co-array. It will be shown in Section 2.5 that the nested array is extremely easy to construct and it is possible to provide exact closed form expressions for the sensor locations and the degrees of freedom for a given number (N) of sensors, unlike the so called Minimum Redundancy Arrays (MRAs) used in [149, 1, 2]. We shall also propose a novel spatial smoothing based technique in Section 2.6 which utilizes only second order statistics to exploit the degrees of freedom offered by the array and works well even for stationary sources. Our technique constructs a suitable covariance matrix (which we shall refer to as the spatially smoothed matrix) corresponding to a longer array on which subspace based methods can be applied directly to perform detection and estimation of more sources than sensors.

As another potential application of the signal model based on the nested array, a new approach to beamforming is proposed in Section 2.7, which directly makes use of the degrees of freedom offered by the co-array. This approach spatially filters the *power* of the sources (instead of their amplitude) and hence it is inherently non-linear in nature. Another major advantage of the proposed approach to beamforming is that, assuming perfect knowledge of the signal covariance matrix, it can eliminate noise, which is never possible using conventional linear approach to beamforming. It should be pointed out that the proposed spatial smoothing based method as well as the beamforming are applicable to any array whose difference co-array is a filled ULA and hence they can be applied even to MRAs.

Relation to Past Work: In earlier works, the problem of detecting more sources than sensors has been addressed in different ways. In [149],[150], it was shown by using minimum redundancy

arrays [126] and an augmented covariance matrix, degrees of freedom can be improved. However, the constructed augmented covariance matrix is not positive semidefinite for finite number of snapshots (and hence violates the condition for being a covariance matrix.) Unlike the augmented matrix in [149], the spatially smoothed matrix proposed in this thesis is *guaranteed to be positive semidefinite for any finite number of snapshots*. In [1],[2], a transformation of the augmented matrix into a suitable positive definite Toeplitz matrix was suggested and an elaborate algorithm was provided to construct this matrix. However, there are two issues with this approach. To gain more degrees of freedom required for detection of more then $N-1$ sources with N sensors, they rely on the class of Minimum Redundancy Arrays (MRAs), for which, unfortunately, there is no closed form expression for either the array geometry or the achievable degrees of freedom for a given N. The optimum design of such arrays is not easy and in most cases, they are restricted to computer simulations or complicated algorithms for sensor placement [96],[110],[148],[159],[39]. Also, the algorithm for finding the suitable covariance matrix corresponding to the longer array is a lengthy and complicated iterative algorithm, which converges only to a local optmimum [1, 2]. In [154], the use of fourth order cumulants was suggested to completely remove the Gaussian noise term and perform better DOA estimation. It was later shown that through the use of fourth order cumulants, one can also achieve significant increase in degrees of freedom [63],[42],[43]. But one weakness of this approach is that it is restricted to non Gaussian sources. Recently, using the concept of Khatri-Rao (KR) product and assuming quasi stationary sources, it has been shown that one can identify upto $2N-1$ sources using a N element ULA [118] without computing higher order statistics. It is to be noted that using the augmented array approach of [149] or with the construction of suitable positive definite Toeplitz matrices as done in [1], it will not be possible to obtain this many degrees of freedom using a ULA. Unfortunately, this method based on quasi stationary sources, is not applicable to stationary signals. In [16], degrees of freedom were increased by generating a virtual sum co-array using a MIMO radar. However, the generation of the sum co-array needs active sensing, i.e., both transmit and receive antennas, and is not applicable to the case of passive sensing.

The content of this chapter is mainly drawn from [135] and portions of it have been presented in [133, 134].

2.2 Outline

In Sec. 2.3, a new signal model based on the concept of a difference co-array is developed. In Sec. 2.4, we discuss the role of difference co-array of the physical antenna array which plays an important role for resolving $O(N^2)$ sources using N sensors. In Sec. 2.5, we propose a new array geometry, namely, the nested array and optimize the degrees of freedom offered by it under a fixed budget of available physical sensors. In Sec. 2.6, we propose a new subspace based algorithm that can fully exploit the structure of nested array and identify $O(N^2)$ sources. In Sec. 2.7, new approaches to deterministic and MVDR-like beamforming are proposed which directly make use of the degrees of freedom offered by the nested array. In Section 5.5, simulations are performed to verify all the proposed techniques and compare them against the conventional methods.

2.3 Definitions and Signal Model based on Difference Co-array

Consider a N element possibly non uniform linear antenna array. Let $\mathbf{a}(\theta)$ be the $N \times 1$ steering vector (see Sec. 1.1.1.1) corresponding to the direction θ whose ith element is $e^{j\frac{2\pi}{\lambda}d_i \sin\theta}$. The sensors are assumed to be placed on a linear grid with d_i denoting the position of the ith sensor. The carrier wavelength is denoted by λ. Let us assume D narrowband sources impinging on this array from directions $\{\theta_i, i = 1, 2, \cdots, D\}$ with powers $\{\sigma_i^2, i = 1, 2, \cdots, D\}$ respectively. Hence the received signal is

$$\mathbf{x}[k] = \mathbf{A}\mathbf{s}[k] + \mathbf{n}[k] \tag{2.1}$$

where $\mathbf{A} = [\mathbf{a}(\theta_1) \quad \mathbf{a}(\theta_2) \cdots \mathbf{a}(\theta_D)]$ denotes the array manifold matrix and $\mathbf{s}[k] = [s_1[k] \quad s_2[k] \cdots s_D[k]]^T$ denotes the source signal vector. The noise $\mathbf{n}[k]$ is assumed to be temporally and spatially white, and uncorrelated from the sources. We also assume the sources to be temporally uncorrelated so that the source autocorrelation matrix of $\mathbf{s}[k]$ is diagonal. Then,

$$\mathbf{R}_{\mathbf{xx}} = E[\mathbf{x}\mathbf{x}^H] \tag{2.2}$$

$$= \mathbf{A}\mathbf{R}_{\mathbf{ss}}\mathbf{A}^H + \sigma_n^2\mathbf{I} \tag{2.3}$$

$$= \mathbf{A} \begin{pmatrix} \sigma_1^2 & & & \\ & \sigma_2^2 & & \\ & & \ddots & \\ & & & \sigma_D^2 \end{pmatrix} \mathbf{A}^H + \sigma_n^2 \mathbf{I} \tag{2.4}$$

Now, following [118], we vectorize $\mathbf{R_{xx}}$ to get the following vector

$$\mathbf{z} = \text{vec}(\mathbf{R_{xx}}) \tag{2.5}$$

$$= \text{vec}\big[\sum_{i=1}^{D} \sigma_i^2 (\mathbf{a}(\theta_i)\mathbf{a}^H(\theta_i))\big] + \sigma_n^2 \vec{\mathbf{1}}_n \tag{2.6}$$

$$= (\mathbf{A}^* \odot \mathbf{A})\mathbf{p} + \sigma_n^2 \vec{\mathbf{1}}_n \tag{2.7}$$

where $\mathbf{p} = [\sigma_1^2 \quad \sigma_2^2 \cdots \sigma_D^2]^T$ and $\vec{\mathbf{1}}_n = [\mathbf{e_1}^T \quad \mathbf{e_2}^T \cdots \mathbf{e_N}^T]^T$ with $\mathbf{e}_i$ being a column vector of all zeros except a 1 at the ith position. Comparing it with (2.1), we can say that $\mathbf{z}$ in (3.17) behaves like the received signal at an array whose manifold is given by $\mathbf{A}^* \odot \mathbf{A}$ where $\odot$ denotes the KR product. The equivalent source signal vector is represented by $\mathbf{p}$ and the noise becomes a deterministic vector given by $\sigma_n^2 \vec{\mathbf{1}}_n$. The distinct rows of $\mathbf{A}^* \odot \mathbf{A}$ behave like the manifold of a (longer) array whose sensor locations are given by the distinct values in the set $\{\vec{\mathbf{x}_i} - \vec{\mathbf{x}_j}, 1 \leq i, j \leq N\}$ where $\vec{\mathbf{x}_i}$ denotes the position vector of the ith sensor of the original array. This array is precisely the difference co-array of the original array [89]. Hence instead of (2.1), we can apply DOA estimation to the data in (3.17) and work with the difference co-array instead of the original array. The difference co-array thus occurs naturally in problems involving second order statistics of the received signal. Even the methods based on fourth order cumulants end up exploiting the degrees of freedom of the difference co-array as was pointed out in [63],[42],[43].

2.4 The Co-Array Perspective

In this section, we shall first discuss the concept of co-array along with its important properties. We shall also relate it to the degrees of freedom for direction of arrival estimation and explain to what extent different algorithms can exploit them.

Definition 2.1: (Difference Co-array)

Let us consider an array of N sensors, with $\vec{\mathbf{x}_i}$ denoting the position vector of the ith sensor. Define the set

$$D = \{\vec{\mathbf{x}_i} - \vec{\mathbf{x}_j}\}, \quad \forall i, j = 1, 2, \cdots, N \tag{2.9}$$

In our definition of the set D, we allow repetition of its elements. We also define the set D_u which consists of the distinct elements of the set D. Then, the *difference co-array* of the given array is defined as the array which has sensors located at positions given by the set D_u. □

The number of elements in the difference co-array (given by the set D_u) directly decides the distinct values of the cross correlation terms in the covariance matrix of the signal received by an antenna array. Using these distinct cross correlation terms judiciously in different ways, one can substantially increase the number of sources that can be detected by the array. Each such technique actually amounts to using a part or whole of the resulting difference co-array, instead of the original array, to perform the DOA estimation. Of course, the maximum degrees of freedom that can be attained is limited by the number of elements present in the co-array. There is a related concept of "sum co-array" which we do not consider here as it does not arise naturally in computation of the covariance matrix of the received signal.

Definition 2.2: (Weight Function) Define an integer valued function $w : D_u \longrightarrow \mathbb{N}^+$ such that

$w(\vec{\mathbf{d}}) =$ no. of occurences of $\vec{\mathbf{d}}$ in D, $\vec{\mathbf{d}} \in D_u$. □

where $\mathbb{N}^+$ is the set of positive integers.The weight function $w(\vec{\mathbf{d}})$ denotes the number of times $\vec{\mathbf{d}}$ occurs.

2.4.1 Properties of Difference Co-Array:

Let us consider any N element array (possibly non linear). Then the following are true for the weight function $w(.)$ corresponding to its difference co-array:

1. $w(\vec{\mathbf{0}}) = N$
2. $1 \leq w(\vec{\mathbf{d}}) \leq N - 1 \quad \forall \vec{\mathbf{d}} \in D_u \setminus \{\vec{\mathbf{0}}\}$
3. $w(\vec{\mathbf{d}}) = w(-\vec{\mathbf{d}}), \quad \forall \vec{\mathbf{d}} \in D_u$
4. $\sum_{\vec{\mathbf{d}} \in D_u, \vec{\mathbf{d}} \neq \vec{\mathbf{0}}} w(\vec{\mathbf{d}}) = N(N-1)$

Property (1) is due to the fact that $\vec{\mathbf{d}} = \vec{\mathbf{0}}$ occurs whenever $\vec{\mathbf{x}_i} = \vec{\mathbf{x}_j}$ in (2.9) and that happens exactly N times for $i = j = 1, 2, \cdots, N$.

Property (2) can be verified as follows: Say the value $\vec{\mathbf{d}} \neq \vec{\mathbf{0}}$ occurs as the difference between the ith and the jth sensors, i.e., $\vec{\mathbf{d}} = \vec{\mathbf{x}_i} - \vec{\mathbf{x}_j}$. Then, $\vec{\mathbf{d}} \neq \vec{\mathbf{x}_i} - \vec{\mathbf{x}_k}$ for any $k \neq j$. Therefore $\vec{\mathbf{d}}$ can occur *only once* as we consider the set of differences in position between a given sensor (say, ith) and the remaining sensors. So the maximum number of times it can occur is once, for each value of $i = 1, 2, \cdots, N-1$, i.e., it can occur a maximum of $N-1$ times. For example, in a N element ULA with unit spacing, the difference of 1 occurs exactly $N-1$ times.

Property (3) is easily verifiable since whenever $\vec{\mathbf{d}} = \vec{\mathbf{x}_i} - \vec{\mathbf{x}_j}$ occurs, correspondingly we also get the position difference $-\vec{\mathbf{d}} = \vec{\mathbf{x}_j} - \vec{\mathbf{x}_i}$. Hence the values $\vec{\mathbf{d}}$ and $-\vec{\mathbf{d}}$ occur with equal frequency.

Property (4) can be verified as follows: The LHS indicates the sum of occurences of all possible non zero position differences $\vec{\mathbf{d}} \in D_u$. The total number of times all position differences can occur is exactly equal to all possible permutations, taken two at a time from the set $\{\vec{\mathbf{x}_i}, i = 1, 2 \cdots, N\}$, which is equal to $N(N-1)$. ▼▼▼

It is to be noted that the cardinality of D_u for a given array gives the degrees of freedom that can be obtained from the difference co-array associated with that array. However, from property (4) above, we can immediately conclude that the maximum degrees of freedom that can be obtainable from a difference co-array for a N element array with *any* geometry, is

$$\mathrm{DOF}_{\max} = N(N-1) + 1. \tag{2.10}$$

Thus we see that if we use second order statistics, then, by exploiting the degrees of freedom (DOF) of the difference co-array, there is a possibility that we can get $O(N^2)$ degrees of freedom using only $O(N)$ physical elements. $\mathrm{DOF}_{\max}$ gives a global upper bound on the maximum DOF achievable from the difference co-array over all classes of arrays. Also there is a trade-off between the degrees of freedom and the value of the weight function which is clear from Property (4) above. If a difference occurs more than once (i.e., $w(\vec{\mathbf{d}}) > 1$ for some $\vec{\mathbf{d}}$), then it implies a drop in the overall cardinality (and thereby the degrees of freedom) of D_u.

2.4.2 Computing the Weight Function:

Given a linear array with d as the minimum spacing of the underlying grid on which sensors are assumed to be located, define the function $c(m)$ which takes a value 1 if there is a sensor located at md and 0 otherwise. Then the weight function $w(n)$ can be computed as the convolution:

$$w(n) = (c \circledast c^{-})(n). \tag{2.11}$$

where $c^{-}(n) = c(-n)$. As an example, the difference co-array of an N element ULA is another ULA with $2N-1$ elements. The difference co-array of a uniform circular array (UCA) is a set of concentric UCAs with $N(N-1)+1$ degrees of freedom when N is odd. Minimum redundancy arrays form a class of non uniform arrays which have the longest difference co-arrays under the constraint that the difference co-array is a ULA. However, the sensor locations and degrees of freedom of such arrays cannot be computed in closed form for any arbitrary N and they are found through computer search [110],[96].

2.4.3 Comparison of underdetermined DOA estimation techniques from the co-array perspective

1. *Augmented Matrix Approach* [149],[150],[148]: This method exploits the difference co-array of a minimum redundancy array but effectively uses only *one half* of the difference co-array, because the negative differences are used up to make the augmented matrix Hermitian. Hence, it can attain $M/2$ degrees of freedom where M is the cardinality of D_u. Also, in case of ULA, this method *cannot* exploit the additional degrees of freedom provided by the difference co-array of the ULA because it only uses one half of the co-array. So it cannot identify sources beyond the traditional limit of $N-1$ for a N element ULA. The modification to [149] that was suggested in [1], [2] also has the same limitation. They assume that the maximum number of resolvable sources that can be obtained from any N element array is $N(N-1)/2+1$ (which is because they aim at constructing a Hermitian Toeplitz matrix) whereas the actual difference co-array provides twice as many degrees of freedom. So using some other suitable algorithm, there is a possibility that this additional freedom might be exploited and even with a N element ULA, one can resolve $2N-2$ sources (see below).

2. *Fourth order cumulants [154],[42],[43][63]:* Here the effective steering vector becomes $\mathbf{a}_{eff} =$

$\mathbf{a} \otimes \mathbf{a}^*$, as given in [154],[43]. Thus $\mathbf{a}_{eff}$ behaves exactly like the steering vector corresponding to the entire difference co-array of the original array, where each row represents an element of the co-array. Depending on the weight function of the co-array, the row corresponding to the nth element of the co-array occurs $w(n)$ times in $\mathbf{a}_{eff}$. Nevertheless, it captures all degrees of freedom offered by the entire co-array and hence even for an ULA with N sensors, it can resolve upto $2N-2$ sources. However, a drawback of this method is that it is applicable only to cases where the source signals are non-Gaussian. Also, one needs to compute the fourth order cumulants of the received signal which might require large number of snapshots and longer computation time.

3. *KR product based MUSIC:*[118] This recent method exploits quasi stationarity of source signals to construct a full rank matrix whose rank is given by the rank of the effective array manifold $\mathbf{A}_{eff} = \mathbf{A}^* \odot \mathbf{A}$ where $\odot$ denotes the KR product. $\mathbf{A}_{eff}$ behaves like the manifold of the entire difference co-array and hence it is inherently capable of exploiting the full degrees of freedom offered by the difference co-array. The authors considered a N element ULA and provided a Krushkal rank based argument to prove that it can identify upto $2N-2$ sources. However such argument is not necessary because the identifiability condition becomes obvious when we view it in the light of the co-array of a ULA. The advantage of this method is that it uses only second order statistics of the data and hence has no problem in handling Gaussian sources. However, it requires quasi stationarity to ensure that the constructed matrix is full rank.

2.5 The Concept of Nested Array: Degrees of Freedom and Optimization

As we observed in Section 2.4, a key idea behind the ability to resolve more sources than physical sensors, is to use a possibly non uniform array so that its difference co-array has significantly more degrees of freedom than the original array. In [149],[1], classes of Minimum Redundancy arrays (MRAs) were used to achieve this purpose. However, the main problem is that there is no simple way to construct these arrays and it requires extensive computer search [159],[39],[186]. Hence, there is no way to predict the attainable degrees of freedom of MRAs for a given N other than the fact that it is always strictly less than $N(N-1)+1$ [186]. So, much of the work on underdetermined

DOA estimation problems have been performed considering specific examples of MRAs instead of a general class.

In this section, we attempt to provide a solution to this problem by proposing a class of "nested arrays". This array structure, as we shall show, can be generated very easily in a systematic fashion and we can exactly predict degrees of freedom of its co-array for a given N. Moreover, with this class, it is indeed possible to generate $O(N^2)$ degrees of freedom from $O(N)$ physical elements. It is also easy to extend this idea of nesting to higher dimensions. A similar increase in degrees of freedom has been achieved in MIMO radar using the virtual array concept through active sensing. But here we will show that even in the *passive* scenario, it is possible to exploit such nested structures and attain *even more* degrees of freedom compared to the corresponding MIMO radar with *same* number of physical elements. The "two level" nested array as we shall define below, is similar to the array structure originally proposed by Bracewell (see [155]). The combination of the transmitting and the receiving arrays in MIMO radar also has a similar structure. However, we shall generalize the structure to beyond 2 levels to systematically increase the degrees of freedom.

2.5.1 Two Level Nested Passive Array

A two-level nested array is basically a concatenation of two ULAs: inner and outer where the inner ULA has N_1 elements with spacing d_1 and the outer ULA has N_2 elements with spacing d_2 such that $d_2 = (N_1 + 1)d_1$. More precisely it is a linear array with sensors locations given by the union of the sets $S_{inner} = \{md_1, \quad m = 1, 2, \cdots, N_1\}$ and $S_{outer} = \{n(N_1 + 1)d_1, \quad n = 1, 2, \cdots, N_2\}$. Fig.2.1 illustrates the nested array. It is similar to the union of the transmit and the receive arrays of a MIMO radar and uses the same number of sensors. However here we use a single such array for receiving only (passive sensing) instead of active sensing as done in MIMO radar.

Fig. 2.1 also shows the difference co-array of this nested array. Two important observations about the co-array are:

- It is a *filled* ULA with $2N_2(N_1 + 1) - 1$ elements whose positions are given by the set S_{ca} defined as

$$S_{ca} = \{nd_1, n = -M, \cdots, M, \quad M = N_2(N_1 + 1) - 1\} \tag{2.12}$$

- Comparing with the virtual array of the MIMO radar, we find that it has twice as many de-

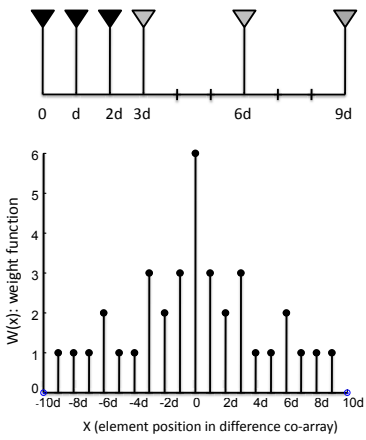


Figure 2.1: A 2 level nested array with 3 sensors in each level (top), and the weight function of its difference co-array (bottom).

grees of freedom though both use the same number of physical sensors. Also, the proposed nested array can be used as a passive array, i.e., the entire array is used for "receive only", and yet we can successfully exploit the degrees of freedom of its co-array (as we shall demonstrate when we perform DOA estimation and beamforming with this array) as opposed to the MIMO radar situation where the longer virtual array can only be realized through "active sensing".

Thus we demonstrated that with a two-level nested array, we can attain $2N_2(N_1+1)-1$ freedoms in the co-array using only $N_1 + N_2$ elements. This indeed gives us a systematic way to increase the degrees of freedom of the co-array, as opposed to the MRAs which need computer search to find those array structures. We can further optimize the distribution of sensors in the two levels by finding N_1, N_2 that maximize the total degrees of freedom, $2N_2(N_1 + 1) - 1$, under the constraint of fixed total number of sensors, i.e., $N_1 + N_2 = N$. The solution (using AM-GM inequalities) can be verified as:

N	optimal N_1, N_2	DOF
even	$N_1 = N_2 = \frac{1}{2}N$	$\frac{N^2-2}{2} + N$
odd	$N_1 = \frac{N-1}{2}, \quad N_2 = \frac{N+1}{2}$	$\frac{N^2-1}{2} + N$

Hence, using 2 level of nesting, we can obtain little over half of the maximum limit in (2.10). So

now we examine how far we can increase the degrees of freedom of the co-array by extending the nesting strategy to more than two levels.

2.5.2 K Levels of Nesting

A 'K-level' nested linear array, parameterized by $K, N_1, N_2, \cdots, N_K \in \mathbb{N}^+$, is defined as one where the sensor positions are given by the set $\mathrm{S}_{(K\text{-level})} = \bigcup_{i=1}^{K} \mathrm{S}_i$ where $\mathrm{S}_i = \{nd \prod_{j=1}^{i-1}(N_j+1), \quad n = 1, 2, \cdots, N_i\}, \quad i = 2, \cdots, K$

$$\mathrm{S}_1 = \{nd, \quad n = 1, \cdots, N_1\}$$

The structure of the ith nesting level ($i \neq 1$) in a K-level nested array is depicted in Fig. 2.2. So, the "K"-level nested array is essentially a union of K ULAs with N_i sensors in the ith level of

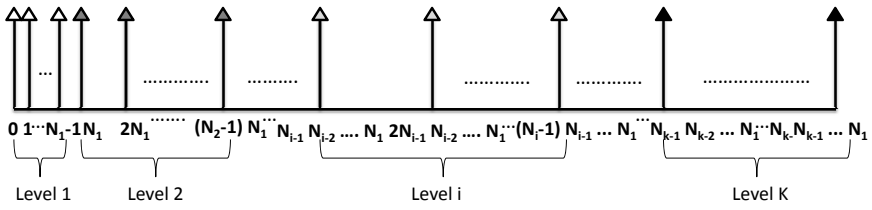


Figure 2.2: The ith nesting level containing N_i sensors in a K-level nested linear array.

nesting and the intersensor spacing in the ith level is $(N_{i-1}+1)$ times the inter-sensor spacing in the $(i-1)$ th level. Note that the total number of physical sensors in the nested array is $\sum_{i=1}^{K} N_i = N$.

It can be verified using the convolution formula in (2.11) that the degrees of freedom in the corresponding difference co-array is $\mathrm{DOF}_K = 2\Big\{[N_2(N_1+1)-1] + [(N_3-1)(N_1+N_2+1)$

$+ (N_1 + 1)] + \cdots + [(N_K - 1)(N_1 + N_2 + N_3 + \cdots + N_{K-1} + 1)$
$+ (N_1 + N_2 + \cdots + N_{K-2} + 1)]\Big\} + 1$
which can be simplified to

$$\mathrm{DOF}_K = 2\Big(\sum_{i=1}^{K}\sum_{j=i+1}^{K} N_i N_j + N_K - 1\Big) + 1 \tag{2.13}$$

The corresponding difference co-array, however, is not a filled ULA for more than two stages of nesting.

2.5.3 Optimization of the K level nested Array

Given a number of sensors, N, we would like to know the number of nested levels and also the number of sensors per nesting level, which will maximize the degrees of freedom of the nested array. This can be cast as the following optimization problem:

$$\max_{K\in\mathbb{N}^+} \max_{N_1,\cdots,N_K\in\mathbb{N}^+} \mathrm{DOF}_K \tag{2.14}$$

$$\text{subject to:} \quad \textstyle\sum_{i=1}^{K} N_i = N$$

Since the variables $K, N_1, \cdots, N_K$ are all positive integers, this is a combinatorial optimization problem. However, we will provide a closed form solution to this problem using a simple observation and thereby deduce the structure of the optimally nested array. The solution to this problem is given by the following theorem

Theorem 2.5.1. *Given a number N of sensors, the optimal number of nesting levels K and the number of sensors per nesting level are given by*

$$K = N - 1$$

and

$$N_i = \begin{cases} 1, & i = 1, 2, 3, \cdots, K-1 \\ 2, & i = K \end{cases}$$

The corresponding difference array is a non uniform linear array with degrees of freedom given by $\mathrm{DOF}_{\mathrm{opt}} = N(N-1) + 1$ *which is same as the upper bound in (2.10).* □

Proof. See Appendix 2.11. □

Corollary 2.5.1. ***Structure of the optimally nested array:*** *Substituting these optimum values into (2.5.2), it can be verified that the* optimum nested array *has sensors located at the positions given by the set* $\mathrm{S}_{\mathrm{opt}} = \{d, 2d, 4d, 8d, \cdots, 2^{N-1}d\}$. □

Hence we have this interesting observation: *The optimally nested array is one with exponential spacing.*

2.5.4 Remarks

1. Compared to the minimum redundancy arrays, our two level nested array will always produce less degrees of freedom since the minimum redundancy arrays are optimized to produce the longest difference co-array under the constraint that the difference co-array is a ULA. However, the structure of a MRA for a given N cannot be exactly predicted and it can only be found through numerical simulations individually for each N. In contrast, the nested arrays are easy to construct and the sensor positions and degrees of freedom are analytically tractable and are also $O(N^2)$.

2. Extending the nesting strategy beyond two levels fails to produce a difference co-array which is a ULA (i.e., holes appear). Same is true for the optimally nested array which is exponentially spaced. This is because the difference array is associated with the second order statistics of the received data. So, extending beyond two levels of nesting will have a nice advantage if we consider higher order statistics. For example, if we consider fourth order statistics, then instead of the difference co-array, we shall be interested in the difference of the difference set. In such a situation, a 4 level nested array with N elements in each level, will beautifully produce a *filled ULA with $O(N^4)$ elements*. Thus, an important advantage of this nesting strategy is that *it easily extends to the case of higher order statistics, producing consistent increase in degrees of freedom*. In future, we shall explore this advantage in greater detail. It is to be noted that MRAs cannot yield $O(N^4)$ degrees of freedom with 4th order statistics because the difference array is a ULA.

3. The optimally nested array belongs to a special class of array, namely, the non redundant array. Non redundant arrays are the ones for which every difference in the difference co-array

appears *only once* (except the zero lag which has to appear N times). Hence, by Property 4 in Section 2.4, it achieves the *maximum value of degrees of freedom* given by (2.10) for any value of N. It is interesting to note that while the two level nested array belongs to the class of "no-hole" arrays (arrays with no holes but with redundancies), the optimally nested array is a "non redundant array" (arrays which have no redundancies but have holes).

For comparison, we enlist the degrees of freedom in the difference co-array of the MRAs [186], 2 level nested array and the upper bound in (2.10) (achieved by the optimally nested array) for different values of N:

N	MRA	2 level	Upper Bound
4	13	11	13
8	47	39	57
9	59	49	73
10	73	59	91
15	159	127	211

2.6 Applications of the nested Array in DOA estimation of more sources than Sensors

As discussed in Section 2.4, the increased degrees of freedom offered by the co-array were exploited by different techniques such as augmented matrix approach [149, 1], fourth order cumulant based methods [154] and the recent quasi stationary signal based method [118]. However, as we pointed out earlier, all of them have limitations. Hence, in the following section, we propose an alternative way to exploit the degrees of freedom by applying the spatial smoothing technique [74] in a novel fashion. This approach guarantees to produce a positive semidefinite covariance matrix corresponding to the co-array even for finite snapshots (thereby overcoming the limitation of [149] without resorting to the multiple iterations suggested in [1]). Also, unlike the methods based on fourth order cumulants, our proposed approach would work even in the case of Gaussian signals and unlike [118], it is equally applicable for stationary signals.

2.6.1 Spatial Smoothing based DOA estimation:

We propose a spatial smoothing based approach for exploiting the degrees of freedom of the difference co-array. It is to be noted that we do not use spatial smoothing for decorrelating uncorrelated sources, as it is done in the conventional sense [74],[163]. Rather, it is used as a technique to build up the rank of an observation matrix so that we do not need to either use the fourth order cumulant or assume quasi stationarity of signals. Note that spatial smoothing works only for a ULA and so we shall focus on the two-level nested array for applying this technique since its difference co-array is a filled ULA. It should be remembered that application of the proposed spatial smoothing based method is not restricted to nested arrays and it can be applied to any array whose difference co-array is a filled ULA (including the MRA).

Consider the signal model in Section 2.3 given by (3.17) for a two level nested array with $N/2$ sensors in each level. We assume N to be even, similar arguments can be applied for the odd case as well. The dimension of the new array manifold $\mathbf{A}^* \odot \mathbf{A}$ is $N^2 \times D$ but it has precisely as many distinct rows as the number of degrees of freedom of the difference co-array which is $(N^2-2)/2+N$ for 2 level nested array for even N. Since the difference co-array in this case is a filled ULA, $\mathbf{A}^* \odot \mathbf{A}$ is like a Vandermonde matrix with $(N^2-2)/2+N$ distinct rows and hence its rank is D as long as $D \leq (N^2-2)/2+N$. However the equivalent source signal vector $\mathbf{p}$ for this co-array consists of the powers σ_i^2 of the actual sources and hence *they behave like fully coherent sources*. In [118], this problem was taken care of by assuming the source powers to vary in different quasi stationary intervals and these were collected to build a full rank matrix. Here, we propose an alternate solution by using the well known technique of spatial smoothing that is applied for coherent sources, to the new signal model given by (3.17). We should like to emphasize that the spatial smoothing technique which is usually applied for ULAs, might also be used for the difference co-array (which is not a ULA) of the optimally nested array after doing some preprocessing like interpolation as suggested by [78].

To apply spatial smoothing, first let us construct a new matrix $\mathbf{A_1}$ of size $((N^2-2)/2+N) \times D$ from $\mathbf{A}^* \odot \mathbf{A}$ where we have removed the repeated rows (after their first occurrence) and also sorted them so that the i th row corresponds to the sensor location $(-N^2/4 - N/2 + i)d$ in the difference co-array of the 2 level nested array. This is equivalent to removing the corresponding rows from the observation vector $\mathbf{z}$ and sorting them to get a new vector $\mathbf{z_1}$ given by

$$\mathbf{z_1} = \mathbf{A_1 p} + \sigma_n^2 \vec{\mathbf{e}}^{'} \tag{2.15}$$

where $\vec{\mathbf{e}}^{'} \in \mathbb{R}^{((N^2-2)/2+N)\times 1}$ is a vector of all zeros except a 1 at the $(N^2/4+N/2)$th position. It can be verified that due to the above mentioned sorting and replacement of repeated rows, the deterministic noise vector changes from $\vec{\mathbf{1}}_n$ in (3.17) to $\vec{\mathbf{e}}^{'}$.

From (2.12), the difference co-array of this 2 level nested array has sensors located from $(-N^2/4-N/2+1)d$ to $(N^2/4+N/2-1)d$. We now divide this co-array into $N^2/4+N/2$ overlapping subarrays, each with $N^2/4+N/2$ elements, where the ith subarray has sensors located at {(-i+1+n)d, n=0,1,$\cdots, N^2/4+N/2-1$} The ith subarray corresponds to the $(N^2/4+N/2-i+1)$th to $((N^2-2)/2+N-i+1)$ th rows of $\mathbf{z_1}$ which we denote as

$$\mathbf{z_{1i}} = \mathbf{A_{1i}p} + \sigma_n^2 \acute{\mathbf{e}}_\mathbf{i}$$

where $\mathbf{A_{1i}}$ is a $(N^2/4+1)\times D$ matrix consisting of the $(N^2/4+N/2-i+1)$th to $((N^2-2)/2+N-i+1)$ th rows of $\mathbf{A_1}$ and $\acute{\mathbf{e}}_\mathbf{i}$ is a vector of all zeros except a 1 at the ith position. It is easy to check that

$$\mathbf{z_{1i}} = \mathbf{A_{11}}\mathbf{\Phi}^{i-1}\mathbf{p} + \sigma_n^2 \acute{\mathbf{e}}_\mathbf{i}$$

where

$$\mathbf{\Phi} = \begin{pmatrix} e^{-j\frac{2\pi}{\lambda}d\sin\theta_1} & & & \\ & e^{-j\frac{2\pi}{\lambda}d\sin\theta_2} & & \\ & & \ddots & \\ & & & e^{-j\frac{2\pi}{\lambda}d\sin\theta_D} \end{pmatrix}$$

Define $\mathrm{R}_i \triangleq \mathbf{z_{1i}}\mathbf{z_{1i}}^H$
$= \mathbf{A_{11}}\mathbf{\Phi}^{i-1}\mathbf{p}\mathbf{p}^H(\mathbf{\Phi}^{i-1})^H\mathbf{A_{11}}^H + \sigma_n^4 \acute{\mathbf{e}}_\mathbf{i}\acute{\mathbf{e}}_\mathbf{i}^H$
$+ \sigma_n^2\mathbf{A_{11}}\mathbf{\Phi}^{i-1}\mathbf{p}\acute{\mathbf{e}}_\mathbf{i}^H + \sigma_n^2\acute{\mathbf{e}}_\mathbf{i}\mathbf{p}^H(\mathbf{\Phi}^{i-1})^H\mathbf{A_{11}}^H$ Taking the average of $\mathbf{R_i}$ over all i, we get

$$\mathbf{R_{ss}} \triangleq \frac{1}{(N^2/4+N/2)}\sum_{\mathbf{i}=1}^{N^2/4+N/2}\mathbf{R_i}. \tag{2.16}$$

We call the matrix $\mathbf{R_{ss}}$ as the spatially smmoothed matrix and it enables us to perform DOA estimation of $O(N^2)$ sources with N sensors, as given by the following theorem:

Theorem 2.6.1. *The matrix* $\mathbf{R_{ss}}$ *as defined in (3.22) can be expressed as* $\mathbf{R_{ss}} = \hat{\mathbf{R}}^2$ *where*

$$\hat{\mathbf{R}} = \frac{1}{\sqrt{N^2/4 + N/2}}(\mathbf{A_{11}}\mathbf{\Lambda}\mathbf{A_{11}}^H + \sigma_n^2\mathbf{I})$$

has the same form as the covariance matrix of the signal received by a longer ULA consisting of $N^2/4 + N/2$ *sensors and hence by applying MUSIC on* $\mathbf{R_{ss}}$*, upto* $N^2/4 + N/2 - 1$ *sources can be identified.* □

Proof. From (3.22), we get $\mathrm{R}_{ss} = \frac{1}{(N^2/4+N/2)} \sum_{\mathbf{i}=1}^{N^2/4+N/2} \mathbf{R_i}$
$= \frac{1}{(N^2/4+N/2)} \Big[\mathbf{A_{11}}\mathbf{C}\mathbf{C}^H\mathbf{A_{11}}^H + \sigma_n^4\mathbf{I}$
$+ \sigma_n^2\mathbf{A_{11}}\mathbf{C} + \sigma_n^2\mathbf{C}^H\mathbf{A_{11}}^H\Big]$ where

$$\mathbf{C} = \begin{pmatrix} \sigma_1^2 & & & \\ & \sigma_2^2 & & \\ & & \ddots & \\ & & & \sigma_D^2 \end{pmatrix} \begin{pmatrix} 1 & \nu_1 & \cdots & \nu_1^{(N^2/4+N/2-1)} \\ 1 & \nu_2 & \cdots & \nu_2^{(N^2/4+N/2-1)} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & \nu_D & \cdots & \nu_D^{(N^2/4+N/2-1)} \end{pmatrix}$$

where

$$\nu_n = e^{-j\frac{2\pi}{\lambda}nd\sin(\theta_i)}$$

Noticing that

$$\mathbf{A_{11}}^H = \begin{pmatrix} 1 & \nu_1 & \cdots & \nu_1^{(N^2/4+N/2-1)} \\ 1 & \nu_2 & \cdots & \nu_2^{(N^2/4+N/2-1)} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & \nu_D & \cdots & \nu_D^{(N^2/4+N/2-1)} \end{pmatrix}$$

we can rewrite (2.6.1) as $\mathrm{R}_{ss} = \frac{1}{N^2/4+N/2}[\mathbf{A_{11}}\Lambda\mathbf{A_{11}}^H\mathbf{A_{11}}\Lambda\mathbf{A_{11}}^H + \sigma_n^4\mathbf{I}$
$+ 2\sigma_n^2\mathbf{A_{11}}\Lambda\mathbf{A_{11}}^H]$

$= \frac{1}{(N^2/4+N/2)}(\mathbf{A_{11}}\mathbf{\Lambda}\mathbf{A_{11}}^H + \sigma_n^2\mathbf{I})^2$ where

$$\mathbf{\Lambda} = \begin{pmatrix} \sigma_1^2 & & & \\ & \sigma_2^2 & & \\ & & \ddots & \\ & & & \sigma_D^2 \end{pmatrix} \tag{2.17}$$

Since $\mathbf{\Lambda}$ is a diagonal matrix with positive entries, the matrix $\hat{\mathbf{R}} \triangleq \frac{1}{\sqrt{N^2/4+N/2}}(\mathbf{A_{11}}\mathbf{\Lambda}\mathbf{A_{11}}^H + \sigma_n^2\mathbf{I})$ has the same form as the conventional covariance matrix used in subspace based DOA estimation techniques when applied on a ULA with $N^2/4 + N/2$ sensors whose array manifold is represented by $\mathbf{A_{11}}$. It shares the same set of eigenvectors as $\mathbf{R_{ss}}$ and its eigenvalues are the square roots of those of $\mathbf{R_{ss}}$. So by eigendecomposition of $\mathbf{R_{ss}}$, the eigenvectors corresponding to the smallest $N^2/4+N/2-D$ eigenvalues of $\hat{\mathbf{R}}$ can be found, which will span the null space of $\mathbf{A_{11}}^H$ and we can apply subspace based algorithms like MUSIC on it to identify upto $D \leq N^2/4+N/2-1$ sources. □

2.6.2 Remarks:

1. The spatial smoothing technique essentially halves the total degrees of freedom offered by the difference co-array, hence we can identify upto $N^2/4 + N/2 - 1$ sources as opposed to $N^2/2 + N - 2$ (if we could exploit total DOF of the full difference co-array). However, using the forward-backward spatial smoothing algorithm, this can be increased to $\frac{3}{2}(N^2/4+N/2-1)$ [151]. The augmented matrix approach [149] or the modified detection estimation algorithm in [1], however, can not identify more than $\frac{1}{2}(N^2/4+N/2)-1$ sources because, as we discussed earlier, they use only one half of the co-array.

2. The co-array of the optimally nested array is not a filled ULA and hence spatial smoothing cannot be used directly on it. It would require some preprocessing like interpolation [78]. However the method based on quasi stationary signals [118] can be applied to it and the *full* degrees of freedom of the optimally nested array can be exploited as we shall demonstrate through numerical examples.

3. Since $\mathbf{R_{ss}}$ is constructed as in (3.22), it is a sum of vector outer products and hence it is *positive semidefinite by construction* for any finite value of snapshots. This essentially overcomes the major flaw of augmented covariance matrix approach [149] without resorting to the iterative

techniques described in [1].

2.7 Beamforming with increased Degrees of freedom

As discussed in the last section, the vector $\mathbf{z}$ in (3.17) can be thought of as the signal received at the difference co-array with the amplitudes of the source signal vector replaced by their corresponding powers. So we can propose beamforming with respect to the signal powers instead of with respect to the signal amplitudes as is the convention. This will enable us to perform beamforming with the *full* degrees of freedom of the difference co-array and gain $O(N^2)$ degrees of freedom using only N physical sensors. To see this, consider taking the inner product of the vector $\mathbf{z}$ with a weight vector $\mathbf{w}$ to get $y = \mathbf{w}^H\mathbf{z}$
$= \sum_{i=1}^{D} \mathbf{w}^H\big(\mathbf{a}^*(\theta_i) \otimes \mathbf{a}(\theta_i)\big)\sigma_i^2 + \sigma_n^2\mathbf{w}^H\vec{\mathbf{1}}_n$ Defining the new beampattern as

$$B_{power}(\theta) = \mathbf{w}^H\big(\mathbf{a}^*(\theta) \otimes \mathbf{a}(\theta)\big), \tag{2.18}$$

we can write (2.7) as

$$y = \sum_{i=1}^{D} B_{power}(\theta_i)\sigma_i^2 + \sigma_n^2\mathbf{w}^H\vec{\mathbf{1}}_n \tag{2.19}$$

Hence the power of the ith source from direction θ_i gets spatially filtered by the amount $B_{power}(\theta_i)$ and thereby spatial filtering (or beamforming) is performed with respect to the power of the signal. As is evident, this kind of beamforming is based on a non linear preprocessing (taking the autocorrelation of the received signal vector) and hence it is essentially a non linear beamformer. It requires time-averaging to realize (3.16) from which the subsequent signal model is derived. It also requires uncorrelated sources and enough snapshots to realize the autocorrelation matrix. An idea similar to this was discussed in [22],[21] where beamforming based on time-averaged arrays was suggested using specific array geometries. We now exhibit some examples of different approaches to beamforming that can be performed with this array. Since the two-level nested array has a uniform difference co-array, we would concentrate on beamforming with 2 level nesting since all the conventional beamforming approaches are based on ULAs. However, the beamforming techniques might be extended to the optimally nested array as well.

2.7.1 Deterministic Beamforming

Let us consider a 2 level nested array with $N/2$ sensors in each level. The difference co-array is an ULA with $(N^2-2)/2+N$ elements. Let $\mathbf{b}(\theta)$ denote the $((N^2-2)/2+N)\times 1$ steering vector of the ULA corresponding to the co-array. Say we want to realize a beam $\mathrm{B}_{des}(\theta)$ with this ULA and let $\mathbf{w}_{des}$ denote the deterministic weights such that

$$\mathbf{w}_{des}^H \mathbf{b}(\theta) = \mathrm{B}_{des}(\theta). \tag{2.20}$$

Now to realize this beampattern with the two level nested array, let $\mathbf{w}$ denote the weight vector we would like to apply to the vector $\mathbf{z}$ in (3.17). Using (2.20), this means

$$\mathbf{w}^H \big(\mathbf{a}^*(\theta) \otimes \mathbf{a}(\theta)\big) = \mathbf{w}_{des}^H \mathbf{b}(\theta), \quad \forall \theta.$$

From the discussion of the co-array of the 2 level nested array in Section 2.5, it can be easily seen that the $N^2 \times 1$ vector $\mathbf{a}^*(\theta) \otimes \mathbf{a}(\theta)$ consists of the same rows as $\mathbf{b}(\theta)$, except the fact that some of them occur more than once. Let n_i denote the number of times the ith row of $\mathbf{b}(\theta)$ occurs in $\mathbf{a}^*(\theta) \otimes \mathbf{a}(\theta)$ and let $i_1, i_2, \cdots, i_{n_i}$ denote the row numbers where it occurs in $\mathbf{a}^*(\theta) \otimes \mathbf{a}(\theta)$. Then the weight vector $\mathbf{w}$ is given by $[\mathrm{w}]_{i_j} = \frac{1}{n_i}[\mathbf{w}_{des}]_i, \quad j = 1, 2, \cdots n_i$
$i = 1, 2, \cdots, (N^2-2)/2+N$

2.7.2 Nulling of Jammers and Noise

Consider J jammers incident on the 2 level nested array from directions $\{\theta_i, i = 1, 2, \cdots, J\}$ respectively. Also assume that we are interested in looking into the direction θ_0. Using the proposed beamformer in (2.20), this means we wish to determine the weight $\mathbf{w}$ to spatially null the jammers and maintain unity response in the look direction. This means

$$B(\theta_0) = 1, \quad B(\theta_i) = 0, \quad i = 1, 2, \cdots, J.$$

However, it can be noticed from (2.19) that the random noise now assumes the form of a deterministic vector $\vec{\mathbf{I}}_{\mathbf{n}}$, which can also be nulled by the weight $\mathbf{w}$. Hence, using this beamforming, we can *effectively null* the noise term as well. This is something which really cannot be achieved by the conventional beamforming. Thus to null both jammers and noise, $\mathbf{w}$ should be selected as the

solution to

$$\begin{pmatrix} \left(\mathbf{a}^*(\theta_0)\otimes\mathbf{a}(\theta_0)\right)^H \\ \left(\mathbf{a}^*(\theta_1)\otimes\mathbf{a}(\theta_1)\right)^H \\ \left(\mathbf{a}^*(\theta_2)\otimes\mathbf{a}(\theta_2)\right)^H \\ \vdots \\ \left(\mathbf{a}^*(\theta_J)\otimes\mathbf{a}(\theta_J)\right)^H \\ \left(\vec{\mathbf{1}}_{\mathbf{n}}\right)^T \end{pmatrix} \mathbf{w} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix} \tag{2.21}$$

With this beamforming, we can null noise, and upto $(N^2-2)/2+N-2$ jammers (with unity response in look direction), since we have in all $(N^2-2)/2+N$ degrees of freedom in the co-array of the 2-level nested array. The jammer DOAs are either assumed known or can be estimated using the proposed spatial smoothing based method in Section 2.6. This technique for nulling jammers can also be extended to the case of the optimally nested array by simply replacing $\mathbf{a}(\theta)$ of the 2 level nested array with that of the optimally nested one in (2.21).

It is to be noted that in practice, since the signal covariance matrix is estimated from a finite number of snapshots, the kronecker product in (2.21) is only an approximation and hence (2.21) is satisfied only approximately. However, as we shall show in the numerical examples, the performance is quite satisfactory for moderate number of snapshots.

2.7.3 MVDR-like Beamforming

When the jammer directions are not specifically known, MVDR beamforming is a popular alternative where the jammers are automatically suppressed by making the weights inversely proportional to the total covariance matrix of signal+jammer+noise while maintaining a distortionless constraint in the look direction [186]. However the MVDR yields poor performance when the jammers are coherent [209]. In our proposed beamforming, though the incident jammers are originally assumed uncorrelated, after the time averaging process, they are represented by their powers which form a deterministic vector $\mathbf{p}$ in (3.17). Hence we cannot apply a MVDR directly on $\mathbf{z}$ since the resulting covariance matrix will be of rank 1. So instead, we perform spatial smoothing on $\mathbf{z}$ as described in the last section by considering the sub-array with elements at $\{nd, n=0,1,\cdots,N^2/4+N/2-1\}$ as the reference sub array (whose steering vector we denote as $\mathbf{a}_1(\theta)$) and get the spatially smoothed covariance matrix $\mathbf{R}_{ss}$ as in (2.6.1). This will help us in building a full rank covariance matrix

which can be effectively used for MVDR beamforming. With constraint of unity response in the look direction θ_0, we can express the MVDR weight as

$$\mathbf{w}_{MVDR} = \frac{\mathbf{R}_{ss}^{-1}\mathbf{a}_1(\theta_0)}{\mathbf{a}_1(\theta_0)^H\mathbf{R}_{ss}^{-1}\mathbf{a}_1(\theta_0)} \tag{2.22}$$

2.8 Numerical Examples

In this section, we provide numerical examples to illustrate the superior performance of the nested array in terms of degrees of freedom for DOA estimation and beamforming. In the examples, we consider a 6 sensor array ($N = 6$) with 8 narrowband sources ($D = 8$) impinging on it from directions of arrival $\{-60°, -45°, -30°, 0°, 15°, 30°, 45°, 60°\}$, all with equal power. The sources are modeled as random Gaussian processes. For comparing with the KR product based MUSIC [118] which requires quasi stationarity, we assume 16 different quasi stationary intervals with T_1 snapshots in each interval. The noise is assumed to be spatially and temporally white. For being able to apply spatial smoothing, we consider a 2 level nested array with 3 sensors in each level. The method based on spatial smoothing can resolve upto $N^2/4+N/2-1 = 11$ sources. The KR-product based method can resolve $(N^2 - 2)/2 + N - 1 = 22$ sources but requires quasi-stationary signals and a larger number of snapshots compared to the proposed method. In particular, if the number of sources is D, then the KR-product based method requires at least D times more snapshots than the proposed spatial smoothing based method.

2.8.1 MUSIC spectra

Fig. 2.3 shows the representative MUSIC spectra after applying the proposed spatial smoothing based technique (SS-method) and the quasi stationary based techniques (QS-method). Both use a total of $T = 4800$ snapshots with the KR product based MUSIC using $T_1 = 300$ snapshots for each of the 16 quasi stationary intervals. The SNR for both the methods is assumed to be 0 dB. As can be seen, both of them can resolve the 8 sources sufficiently well.

Now let us reduce the total number of snapshots to $T = 480$ snapshots. Fig. 2.4 illustrates the MUSIC spectra for the proposed SS-method and the QS-method. The QS method clearly suffers

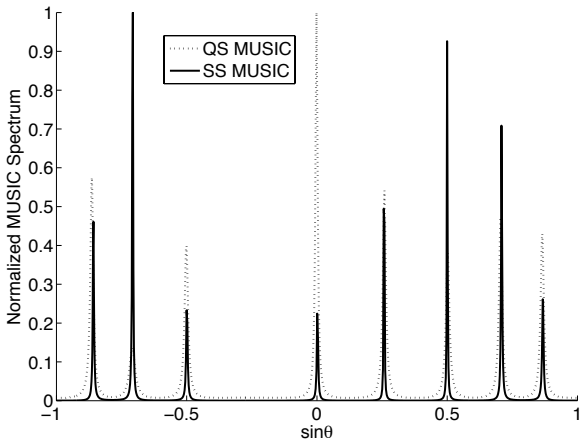


Figure 2.3: MUSIC spectrum using the SS-method and the QS-method, as a function of sine of the DOA, $N = 6$, $D = 8$, $T = 4800$, SNR$= 0$ dB.

due to lack of snapshots and it misses the peak at -60° whereas the SS method still shows a good performance with clearly discernible peaks of the MUSIC spectrum. This shows the effect of snapshots on the performance of the two methods. Next, we study the performance of both the methods through Monte Carlo simulations.

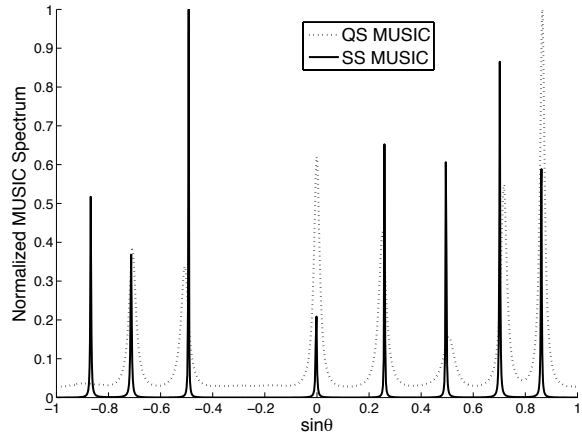


Figure 2.4: MUSIC spectrum using the SS-method and QS-method, as a function of sine of the DOA. Here $N = 6$, $D = 8$, $T = 480$, SNR$= 0$ dB.

2.8.2 RMSE v/s SNR and snapshots

We compare the SS-method and QS-methods by studying the RMSE of the angle estimates as a function of both SNR and snapshots. Since the two level array has 12 degrees of freedom, we also

consider the corresponding RMSE for conventional MUSIC applied to a 12 element ULA which we would use as a benchmark at high SNR. We plot the RMSE for the source at 30°. The performance is similar for the other sources as well. Fig. 2.5 shows the RMSE of the three methods as a function of SNR, averaged over 1500 Monte Carlo simulations, for $T = 800$ snapshots. Fig. 2.6 show the corresponding performances for $T = 4800$ snapshots.

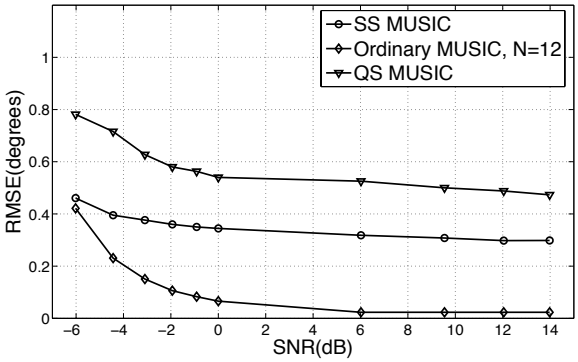


Figure 2.5: RMSE (in degrees) vs. SNR (for the source at 30°) of QS and SS methods applied on 2 level nested array and ordinary MUSIC on a 12 element ULA, with $T = 800$, $N = 6$.

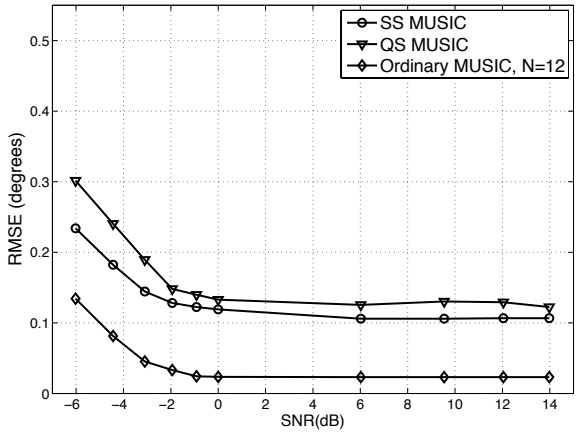


Figure 2.6: RMSE (in degrees) vs. SNR (for the source at 30°) of QS and SS methods applied on 2 level nested array and ordinary MUSIC on a 12 element ULA, with $T = 4800$, $N = 6$.

Notice how the performance of both SS and QS methods improve with SNR. Also, the SS method performs reasonably better than the KR method at low snapshots whereas the difference in their performance becomes less prominent at larger number of snaps. Note, both the methods perform worse than the conventional MUSIC applied to a 12 element ULA because of the finite

sample size due to which the kronecker product in (2.18) is only an approximation.

Next, in Fig. 2.7 we plot the performance of the three methods for SNR=6 dB by varying the total number of snapshots T. The performance of both SS and QS improve considerably with the number of snapshots with the QS method performing strictly worse than the proposed SS method.

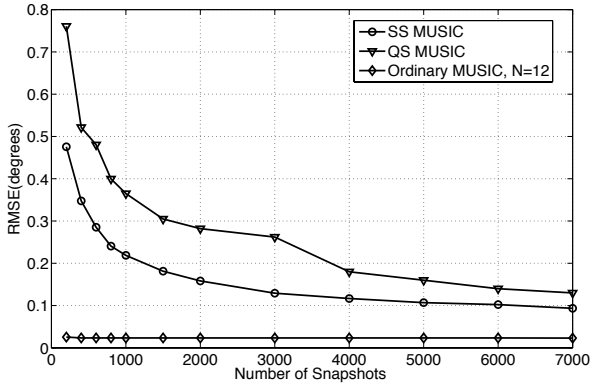


Figure 2.7: RMSE (in degrees) vs. the number of snapshots T (for the source at 30°) of QS and SS methods applied on 2 level nested array and ordinary MUSIC on a 12 element ULA. Here SNR$= 6$ dB, $N = 6$.

2.8.3 Detection Performance

In the previous examples, we assumed the number of source D to be known. However, in practical situations, we need to estimate it from the available data. To do this, we would need to find some optimum detection algorithm suited to our model, which is a subject of our future research in this direction. However, we applied the threshold based technique in [41] which proved to give satisfactory results in the proposed SS based technique. We plot the estimated probability of detection of this proposed method at SNR$= 10$ dB as a function of snapshots in Fig. 3.21. For comparison, we also plot the corresponding performance of the conventional MUSIC applied on 12 element ULA. The probabilities were obtained by running 1000 Monte Carlo simulations averaged over another 1000 runs. We can see that the detection performance improves dramatically with increasing snapshots and approaches that of the conventional MUSIC asymptotically.

2.8.4 Resolution Performance

The advantage of MUSIC-like algorithms comes from the fact that these are super-resolution algorithms, capable of resolving even very closely spaced sources. Since SS-MUSIC makes use of the

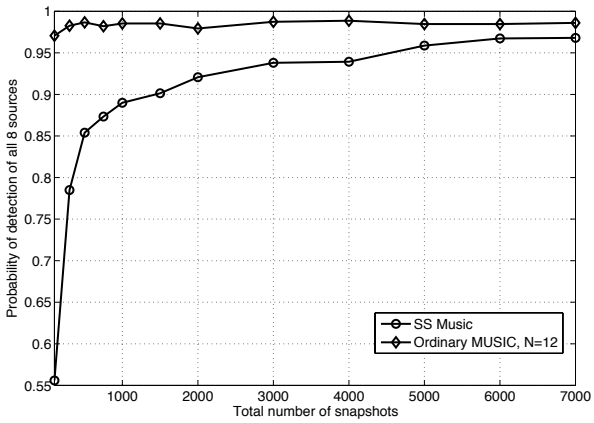


Figure 2.8: Comparison of detection performance of SS method applied to 2 level nested array and MUSIC applied to 12 element ULA as a function of number of snapshots. Here SNR$=10$ dB, $N=6$.

increased degrees of freedom provided by the co-array, we expect its resolution performance to improve greatly. In this example, we consider two sources placed at 12° and 14° and compare the resolution performances of two-level nested array with 6 sensors, an ULA with 6 sensors and an ULA with 12 sensors (since SS MUSIC applied on 6 element ULA yields 12 degrees of freedom). Following [98], we plot the probability of resolution vs SNR for $T=300$ snapshots, averaged over 1000 Monte Carlo runs in Fig 2.9. As in [98], we assume that the number of sources is known. Clearly the two level nested array outperforms the corresponding ULA with same number of sensors and performs close to the much longer ULA.

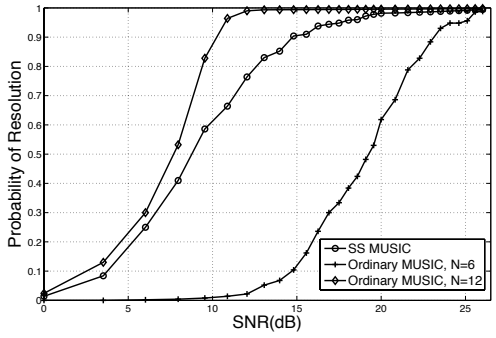


Figure 2.9: Comparison of resolution performance of SS-MUSIC method applied to 2 level nested array with 6 sensors and traditional MUSIC applied to a 12 element ULA and a 6 element ULA, as a function of SNR for two closely spaced sources at 12° and 14°.

2.8.5 2 level vs optimally nested array

We now provide an example to show the ability of the optimally nested array to resolve more sources than the corresponding 2 level array for $N = 6$ sensors. Since the spatial smoothing based technique cannot be applied directly to the optimally nested array, we consider QS method to perform the DOA estimation. With QS method, the 2 level array can resolve $D = 22$ sources whereas the optimally nested array can resolve 30 sources. We consider 27 quasi stationary sources distributed uniformly between -60° and 60°. We use $2 \times 27 = 54$ quasi stationary intervals with $T_1 = 300$ snapshots per interval yielding a total of $54 \times 300 = 16200$ snapshots. Fig. 2.10 and Fig. 2.11 respectively show the corresponding spectrum obtained by using QS-MUSIC to the optimally nested array and the 2 level nested array. Clearly the 2 level nested array fails to detect the sources whereas the optimally nested array can perfectly detect them. This confirms the superior detection ability of the optimally nested array. For large N, the optimally nested array can detect about twice as many sources as the 2 level nested array.

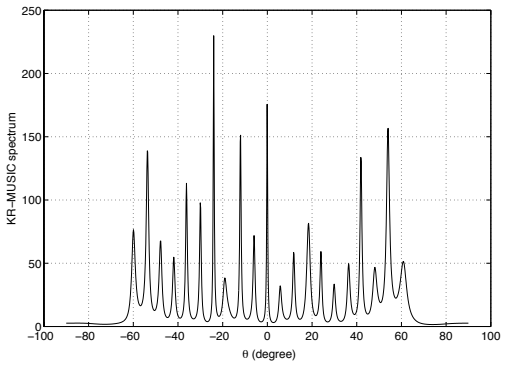


Figure 2.10: MUSIC spectrum for QS-method applied to the optimally nested array with $N = 6$, $T = 16200$, $D = 27$ and SNR$= 0$ dB.

2.8.6 Beamforming

In this section we demonstrate the capability to perform beamforming with increased degrees of freedom offered by the 2 level nested array. There are two ways of plotting the resultant beampattern. One way is to plot the pattern $B(\theta)$ as defined in (2.18) directly after computing the necessary weights. This however is not practical because the kronecker product in (2.18) is realized only

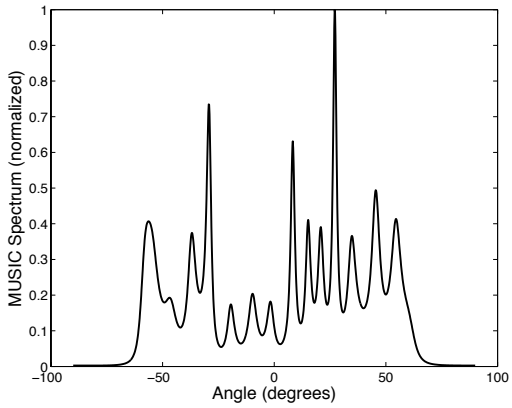


Figure 2.11: MUSIC spectrum for QS-method applied to the 2 level nested array with $N = 6$, $T = 16200$, $D = 27$ and SNR$= 0$ dB.

approximately due to finite snapshots. Hence, for practical reasons, to plot a more realistic beampattern, we consider a single source at angle θ and plot the amplitude response of our non linear beamformer to it by varying θ for *finite* number of snapshots. This automatically takes into consideration the finite sample effect. We call the former beam as an ideal beampattern and the latter as the practical beampattern.

2.8.6.1 Deterministic Beampattern

Let us consider a simple sinc beampattern obtained by applying uniform weighting to a 19 element array. Since $N = 6$ element array with 2 level of nesting gives $(N^2 - 2)/2 + N = 23$ degrees of freedom, we consider a 6 element array and plot both the ideal and practical beampatterns in Fig. 2.12 by computing the weights following (2.7.1). The practical beampattern is generated using $T = 100$ snapshots and it approximates the ideal beampattern quite closely.

2.8.6.2 Jammer and noise Nulling

Consider a 4 element array with 2 levels of nesting. The look direction is 0° and we consider 6 jammers from directions $\{-60^\circ, -45^\circ, -30^\circ, 15^\circ, 30^\circ, 45^\circ\}$ with jammer to signal power ratio 20 dB and signal to noise ratio 0 dB. The ideal and practical beampatterns are generated using the weight computed from (2.21) and plotted in Fig 2.14. Note the introduction of nulls along the jammer positions. Also, the sidelobes are seen to be high which could pose problems for noise in the

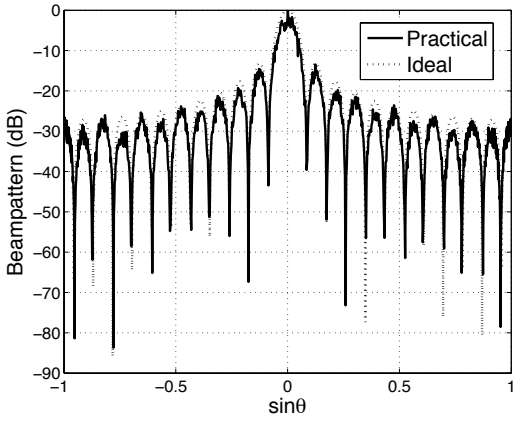


Figure 2.12: Practical deterministic sinc beampattern obtained by non linear beamforming from the 2 level nested array, $N = 6$, $T = 100$.

conventional beamforming. However, unlike conventional beamforming, in our case the noise has been reduced to a deterministic vector which can also be efficiently nulled. So the presence of noise at all spatial angles (which is assumed in conventional beamforming) can be ruled out and hence the high sidelobes do not pose a serious threat as far as noise is concerned. Also, we see that though this jammer and noise nulling theoretically requires perfect knowledge of signal covariance matrix, the practical beampattern shows that the approximation is reasonably good at moderate number of snapshots ($T = 100$).

In order to show the performance improvement of our proposed beamforming, we plot the signal to jammer plus noise ratio (SJNR) vs the number of snapshots in Fig 2.13. The SJNR plot is obtained as follows. The response of a beamformer to an input signal consisting of desired signal+jammer+noise is measured. Then the response of the beamformer to the input signal consisting only of the same jammer+noise is measured. Finally the ratio between the difference of the first and second responses, and the second response is taken to be the SJNR and it is plotted on a dB scale, averaged over 1000 Monte Carlo simulations. Notice how the SJNR performance of proposed method improves with number of snapshots. We also compare it with the SJNR performance of an equivalent ULA with 11 sensors (since the co-array of a 2 level nested array with 4 sensors has 11 degrees of freedom) where we introduced nulls along jammer directions and used remaining degrees of freedom to maximize the SNR. As expected, the SJNR of the equivalent ULA is not affected by number of snapshots. Also, it is to be noted that the proposed beamforming outperforms the ULA. This can be explained from the fact that we are actually able to null the noise (the extent to

which it is nulled again depends on snapshots) which is not possible for the ordinary beamforming case with the equivalent ULA, and hence the SJNR for our method increases almost linearly (on dB scale) with the number of snapshots.

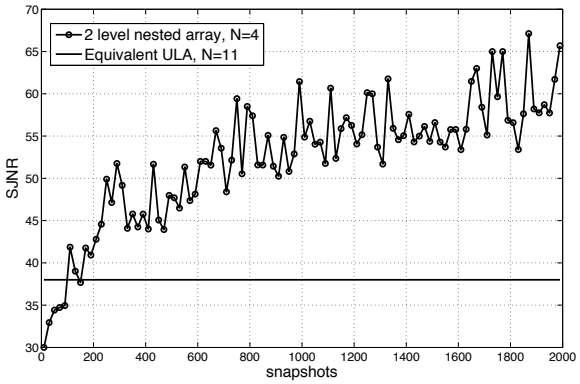


Figure 2.13: Signal-to-Jammer-and-noise-ratio (SJNR) vs number of snapshots (averaged over 1000 Montecarlo runs) for 2 level nested array with 4 sensors and the equivalent ULA with 11 sensors, Number of jammers $= 6$, SNR$= 0$ dB, SJR$= -20$ dB.

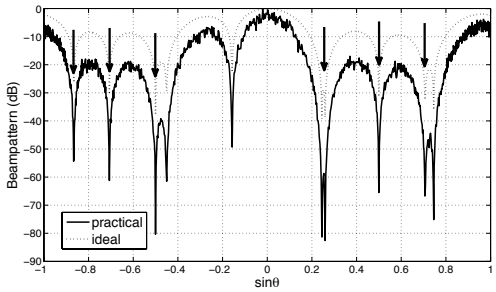


Figure 2.14: Practical Beampattern obtained from the 2 level nested array after nulling 6 jammers and noise, $N = 4$, $T = 100$.

2.8.6.3 MVDR-like Beampattern

We now plot the MVDR-like beampattern for the spatially smoothed 6 element array with 2 levels of nesting. Look direction is 0° and we consider 9 jammers at $\{ -60^\circ, -45^\circ, -30^\circ, -15^\circ, 20^\circ, 30^\circ, 45^\circ, 55^\circ, 60^\circ \}$. The weights are computed as per (2.22) using $T = 1000$ snapshots to compute the spatially smoothed covariance matrix $\mathbf{R}_{ss}$, and the ideal beampattern is plotted in Fig 2.15.

Through the inversion of the spatially smoothed matrix in (2.22), nulls have been automatically introduced in the jammer directions even without explicitly finding the jammer DOAs.

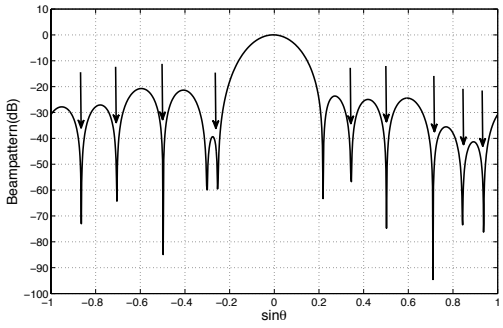


Figure 2.15: Ideal MVDR-like beampattern obtained by applying spatial smoothing to the 2 level nested array, $N = 6$, using $T = 1000$ snapshots for computing the smoothed covariance matrix.

2.9 Conclusion

In this chapter, we have attempted a unified viewpoint to compare the established approaches to the problem of underdetermined source localization in the light of the co-array concept. Drawing inspiration from the MIMO radar literature, we proposed a novel nested array structure which can realize significantly more degrees of freedom even in the passive sensing scenario. Different nesting strategies were explored and the optimum nested array structure was found through solving a combinatorial optimization problem. Though this nested array could be used in conjunction with both the fourth order cumulant based and the KR-product based methods, it would suffer from the inherent drawbacks of these techniques. So we proposed an alternative spatial smoothing based approach to underdetermined DOA estimation, which does not require any of the assumptions in the previous two methods. We also demonstrated how to perform beamforming with this array structure. The results were verified through simulations and the performances of the proposed techniques are seen to improve considerably by increasing the number of snapshots.

2.10 Appendix

2.11 Proof of Theorem 2.5.1

Using the constraint in (2.14), we can write (2.13) as $\text{DOF}_K = 2\Big\{\frac{1}{2}\Big[\Big(\sum_{i=1}^{K} N_i\Big)^2 - \sum_{i=1}^{K} N_i^2\Big] + N_K\Big\} - 1$

$= N^2 - \sum_{i=1}^{K} N_i^2 + 2N_K - 1$

Let us consider N_j, the number of sensors in the jth level of nesting. We shall deal with the following two cases separately:

- **Case I:** $1 \leq j \leq K-1$. Consider breaking N_j into the sum of two smaller integers N_{j1} and N_{j2}, i.e., $N_j = N_{j1} + N_{j2}$. This amounts to breaking the jth nested level into 2 levels such that now there are $(K+1)$ levels of nesting with $\{ N_1, N_2, \cdots, N_{j-1}, N_{j1}, N_{j2}, N_{j+1}, \cdots, N_K \}$ sensors respectively. Using the expression for degrees of freedom in (2.11), we can write, the new degrees of freedom for this new nested array, as $\text{DOF}_{K+1} = N^2 - \left[\sum_{i \neq j, i=1}^{K} N_i^2 + N_{j1}^2 + N_{j2}^2 \right] + 2N_K - 1$.

  Subtracting (2.11) from (2.11), we get the change in DOF due to splitting the jth nesting level:
  $\triangle\text{DOF} \triangleq \text{DOF}_{K+1} - \text{DOF}_K = N_j^2 - N_{j1}^2 - N_{j2}^2$
  $= 2N_{j1}N_{j2} > 0$

  Thus we can conclude that *breaking up the* jth *level of nesting into two levels, always increases the degrees of freedom for* $j = 1, 2, 3, \cdots, K-1$. Hence we should go on redistributing the sensors in the inner nesting levels till we hit only one sensor per nesting level. So, at this stage we have $\hat{N}_K$ elements in the last stage and all the stages before it have one sensor each, i.e., $N_i = 1, \quad i = 1, 2, 3 \cdots, K-1$ Now we consider the last stage of nesting and optimize the number of sensors in them.

- **Case II:** $j = K$. Similar to the previous case, let us break N_K as $N_K = N_{K1} + N_{K2}$ So now we have a $K+1$ level-nested array, with $N_1, N_2, \cdots, N_{K1}, N_{K2}$ sensors respectively in the $K+1$ levels. Applying the formula for degrees of freedom in (2.11), we get the degrees of freedom in this case as, $\text{DOF}_{K+1,K} = N^2 - \left[\sum_{i=1}^{K-1} N_i^2 + N_{K1}^2 + N_{K2}^2 \right] + 2N_{K2} - 1$ Similarly as before, the increase in degrees of freedom can be found by subtracting (2.11) from (2.11) as
  $\triangle\text{DOF}_{K+1,K} \triangleq \text{DOF}_{K+1,K} - \text{DOF}_K$
  $= N_K^2 - N_{K1}^2 - N_{K2}^2 + 2(N_{K2} - N_K)$
  $= 2N_{K1}N_{K2} - 2N_{K1}$
  $= 2N_{K1}(N_{K2} - 1) \geq 0$

  Thus we see that in this case also, the degrees of freedom increase by splitting as long as the number of sensors in the last level (i.e., N_{K2}) is ≥ 2. So we should go on splitting the last level of nesting till we hit 2 sensors in the last level and one sensor in every other level. This, along with the previous case, also implies that the total number of levels of nesting is $N - 2 + 1 = N - 1$ since the total number of sensors is N.

Chapter 3

Nested Arrays in Two Dimensions

3.1 Introduction

In Chapter 2, one dimensional ($1D$) nested arrays were introduced to provide a novel way to perform array processing with increased degrees of freedom using much fewer physical sensors. In this chapter, we generalize the idea of nested arrays to two dimensions ($2D$) assuming the sensors to be present on lattice(s). The extension of the idea of $1D$ nested array to $2D$ is quite nontrivial, and it involves addressing several issues pertaining to the array geometry, identifiability and so forth, which are peculiar only to the $2D$ case. This chapter puts forward the theory and applications of the class of $2D$ nested arrays, which offers the potential to do conventional $2D$ array processing applications with enhanced degrees of freedom. The goals of this chapter are to study the geometrical structure of nested arrays in $2D$ and propose practical algorithms for exploiting the increased degrees of freedom provided by the nested array in $2D$ DOA estimation, beamforming, and so forth.

In two dimensions, lattices [182] are widely used to perform uniform sampling of signals and are analogous to uniform linear arrays (ULAs) in $1D$. In this chapter, we propose a new class of two dimensional arrays by combining sensors on two (possibly nonseparable) lattices, which is capable of providing a dramatic increase in the degrees of freedom (DOF) compared to the actual number of physical sensors. We call this class of arrays as "nested arrays" because they are obtained by combining two arrays with elements on two related lattices, one dense and the other sparse, the sparse lattice being a sublattice of the dense one.

Traditional subspace based techniques are based on processing the second order statistics of the impinging sources, assuming them to be wide sense stationary (WSS) [160, 186]. In the case of our

proposed nested arrays also, we shall demonstrate that using only the second order statistics of the impinging sources, it is possible to create the effect of a $2D$ array of $O(MN)$ "virtual" sensors using only $M + N$ physical sensors. We want these virtual sensors to be distributed uniformly on a lattice, completely filling up an appropriate parallelogram with no missing sensors (also called "holes"). This is mathematically defined in Sec. 3.3, Definition 3.3. We will henceforth refer to sensors arranged this way as "contiguous sensors" without holes or missing sensors. Unlike $1D$ nested arrays, there are several possible orientations of the co-array for the $2D$ nested array in the $2D$ plane. By manipulating the relative configurations of the sparse and dense arrays, one can further increase the number of virtual sensors in $2D$. These interesting issues relating to the geometric orientation of $2D$ nested arrays are addressed in detail, and the number of virtual sensors is considerably improved by choosing an appropriate orientation.

Relation to Past Work. The main concept behind generation of this larger virtual array is based on the idea of "difference co-arrays" which is discussed in [89, 110] in great detail. In earlier work, the problem of detecting more sources than sensors [1, 2, 149] (underdetermined scenario), has mostly been discussed in the context of $1D$ arrays using non uniform arrays like minimum redundancy arrays (MRA) [126] or their approximations [159],[148]. In [135] and in Chapter 2, the concept of $1D$ nested arrays was introduced, which is generalized to two dimensions in this chapter. In two dimensions, the idea of difference co-array was used earlier in [155] and the references therein. However, they address very specific examples of $2D$ arrays and do not develop the theory of a generic class of $2D$ arrays. Also, they do not develop any algorithm for exploiting the extra degrees of freedom offered by the $2D$ co-array for array processing applications like $2D$ DOA estimation or beamforming. In [184], [185], [136], a related idea of coprime arrays in $1D$ and $2D$ were introduced. Coprime arrays can be physically made arbitrarily sparse but they usually require more physical sensors to produce contiguous virtual sensors on a lattice.

The content of this chapter is mainly drawn from [142] and [143] and parts of it have been presented in [138].

3.2 Outline

In this chapter, we first briefly review $2D$ lattices and the idea of difference co-arrays in Sections 3.3 and 3.4. In Section 3.5, we propose the new class of $2D$ nested arrays and characterize their

degrees of freedom. In two dimensions, construction of these arrays gives rise to several issues pertaining to their geometrical orientations (which we will henceforth refer to as symmetry issues) and these are addressed in Section 3.6. In Section 3.7, we revisit the nested array by considering the Smith form of an appropriate integer matrix that relates the lattice generators of the dense and the sparse lattice. This is shown to offer a new perspective into the problem of array design and symmetry issues. In Section 3.8, the total number of degrees of freedom in the virtual array is optimized under the constraint of a fixed number of sensors in the physical domain. Several interesting conclusions leading to the optimal structure are drawn. In Section 3.9, we propose a novel algorithm, inspired from the idea of $2D$ spatial smoothing [33, 74, 163], which provides a way to build a larger covariance matrix corresponding to the virtual array. In Section 3.10, we address some of the issues related to identifiability of $2D$ source DOAs, which are inherent in any $2D$ array processing problem and hence, applies in particular to the virtual $2D$ array of our proposed nested array as well. Finally in Section 3.11, we provide numerical examples to demonstrate the DOA estimation performance of the proposed algorithm and study its performance with respect to SNR and the number of snapshots.

3.3 Review and Definitions of Multidimensional lattices and Integer Matrices

We shall first review several concepts related to multidimensional lattices [182], with particular reference to integer matrices, which will be extensively used in the following sections.

Given a $D \times D$ non singular matrix $\mathbf{V}$, the D dimensional lattice generated by this matrix is defined as

$$LAT(\mathbf{V}) = \{\mathbf{t} : \mathbf{t} = \mathbf{V}\mathbf{n}\}$$

where $\mathbf{n} \in \mathbb{N}^{D\times 1}$ is an integer vector. The matrix $\mathbf{V}$ is called the generator of the lattice. An important concept related to $LAT(\mathbf{V})$ is that of the *fundamental parallelepiped* (FPD):

Definition 3.1: (**Fundamental Parallelepiped (FPD)**:) The FPD of $\mathbf{V} \in \mathcal{C}^{D\times D}$ (denoted $FPD(\mathbf{V})$)

in D dimensions is defined as the set of all vectors of the form

$$\{\mathbf{V}\mathbf{x}, \mathbf{x} \in [0 \quad 1)^D\}$$

Visually, $FPD(\mathbf{V})$ consists of all points contained in the parallelepiped whose sides are given by the two column vectors of $\mathbf{V}$ (it includes the origin but excludes all other corners). This is illustrated in Fig. 3.1 for the two dimensional case. □

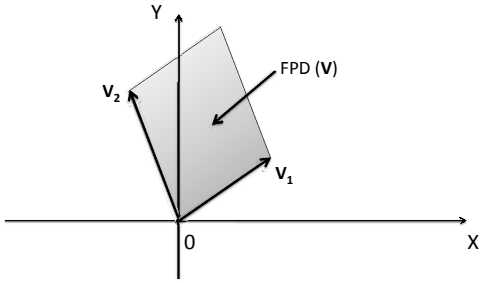


Figure 3.1: Fundamental parallelepiped (FPD) of the lattice generated by the generator matrix $\mathbf{V} = [\mathbf{v}_1 \quad \mathbf{v}_2]$

The volume of the FPD is given by $|\det(\mathbf{V})|$ and the number of lattice points per unit volume (density of lattice points) is given by $1/|\det(\mathbf{V})|$.

A matrix whose elements are all integers, is known as an integer matrix. The unimodular matrix is a special kind of integer matrix whose determinant is ± 1. The inverse of a unimodular matrix is also a unimodular matrix. It can be verified that $LAT(\mathbf{V}) = LAT(\mathbf{V}\mathbf{U})$ for any unimodular matrix $\mathbf{U}$. The number of integer vectors (vectors whose entries are all integers) within the FPD of an integer matrix $\mathbf{P}$, is equal to $\det(\mathbf{P})$. An important result on integer matrices is the Smith-form decomposition [182]:

Definition 3.2: (Smith-form Decomposition): Any integer matrix $\mathbf{A}$ of size $D \times N$ can always be decomposed as

$$\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}$$

where $\mathbf{U}$ and $\mathbf{V}$ are unimodular matrices (of size $D \times D$ and $N \times N$ respectively) and $\mathbf{\Lambda}$ is a

diagonal integer matrix whose ith element λ_i divides the $(i+1)$th element λ_{i+1}. □

In this context, we shall also like to mathematically defined the notion of contiguous sensors in $2D$ which will come up several times in this chapter.

Definition 3.3: (Contiguous sensors on a lattice): In two dimensions, a collection of sensors on $LAT(\mathbf{V})$ is said to be contiguous (also referred to as "array with no holes", and "array with no missing sensors") if the sensor locations are given by

$$\{\mathbf{V}[v_1 \quad v_2]^T, N_{11} \leq v_1 \leq N_{12}, N_{21} \leq v_2 \leq N_{22}\} \tag{3.1}$$

for some integers $N_{11}, N_{12}, N_{21}, N_{22}$. □

3.4 Revisiting Difference Co-Arrays

The role of difference co-array in $1D$ in the context of $1D$ nested arrays has been discussed in detail in Chapter 2. The case of $2D$ difference co-arrays in the context of $2D$ coprime arrays, was elaborated in [185]. In this section, we define a related concept of cross difference co-array is also explicitly pointed out as follows:

Cross Difference Co-array: Given two arrays, one with N_1 sensors with locations $\{\vec{\mathbf{x}}_{1i}, 1 \leq i \leq N_1\}$ and the other with N_2 sensors with locations $\{\vec{\mathbf{x}}_{2i}, 1 \leq i \leq N_2\}$, their "cross difference co-array" is defined as the set

$$\pm\{\vec{\mathbf{x}}_{1i} - \vec{\mathbf{x}}_{2j}, 1 \leq i \leq N_1, 1 \leq j \leq N_2\}.$$

Notice that the definitions of difference co-array and cross difference co-array apply to position vectors in any dimension. The difference co-array arises naturally when the second order moments of the signals impinging on a sensor array are considered. Assume a narrowband signal $s(t)$ with wavelength λ impinging on a two dimensional array with sensors located at $\{\vec{\mathbf{v}}_n \in \mathbb{R}^{2\times 1}, n = 1, 2, \cdots, N\}$. Let ϕ denote the angle of elevation and θ denote the azimuthal angle respectively. Then, the signal received at the nth sensor is given by

$$x_n(t) = e^{j\frac{2\pi}{\lambda}\sin\phi\vec{\mathbf{u}}^T\vec{\mathbf{v}}_n}s(t)$$

where $\vec{\mathbf{u}} = [\cos\theta \quad \sin\theta]^T$. Assuming the power of $s(t)$ is σ_s^2, the cross correlation between the signals received at the mth and nth sensors is given by

$$E(x_m(t)x_n^*(t)) = e^{j\frac{2\pi}{\lambda}\sin\phi\vec{\mathbf{u}}^T(\vec{\mathbf{v}}_m-\vec{\mathbf{v}}_n)}\sigma_s^2.$$

The difference co-array elements $\vec{\mathbf{v}}_m - \vec{\mathbf{v}}_n$ appear in the exponents of the cross correlation terms and these phases behave exactly like those received by (virtual) sensors located in the (larger) difference co-array. We state two properties of difference co-arrays in this context which are particularly useful:

1. If $\mathbf{v}$ denotes the position of a virtual sensor in the difference co-array, then there exists a corresponding virtual sensor in the co-array at $-\mathbf{v}$ (image of $\mathbf{v}$ under point reflection about $\mathbf{0}$).

2. The maximum possible number of virtual sensors in the difference co-array of an array with N sensors (with any geometry and any dimension) is $N(N-1)+1$ [135].

Hence, if the N physical sensors are chosen appropriately one can generate a difference co-array with as many as $N(N-1)+1$ virtual sensors. It is to be noted that the difference co-array is symmetric under point reflection about $\mathbf{0}$. In $2D$, since there are 4 quadrants, there is room to play around with the geometric orientation of the physical array so that the resulting difference co-array has a desired distribution of sensors among the quadrants. We shall explore this idea in detail in Sec 3.6.

Since the difference co-array arises while computing the autocorrelation of the received signal, we can get the effect of a larger array by constructing a signal model based on the second order moments and can think of performing array processing with *increased degrees of freedom* by using the difference co-array. In this chapter, we aim to achieve this by first constructing a class of $2D$ arrays with $M+N$ sensors distributed over two non separable lattices, such that its difference co-array has $O(MN)$ sensors on a lattice with no holes (missing sensors) inside. This is analogous to a uniform linear array (ULA) in one dimension.

3.5 Nesting of Arrays on Lattices

Consider two lattices with 2×2 generator matrices $\mathbf{N}^{(\mathrm{s})}$ and $\mathbf{N}^{(\mathrm{d})}$ (not necessarily integer matrices) satisfying

$$\mathbf{N}^{(\mathrm{s})} = \mathbf{N}^{(\mathrm{d})}\mathbf{P} \tag{3.2}$$

where $\mathbf{P}$ is a 2×2 integer matrix. Notice:

- $LAT(\mathbf{N}^{(\mathrm{s})})$ is a sublattice of $LAT(\mathbf{N}^{(\mathrm{d})})$, i.e., if $\mathbf{v} \in LAT(\mathbf{N}^{(\mathrm{s})})$ then $\mathbf{v} \in LAT(\mathbf{N}^{(\mathrm{d})})$. Hence, $LAT(\mathbf{N}^{(\mathrm{s})})$ is a sparser lattice compared to $LAT(\mathbf{N}^{(\mathrm{d})})$. This is illustrated in Fig. 3.2 where $\mathbf{P} = \begin{pmatrix} 2 & -2 \\ 2 & 1 \end{pmatrix}$ and $\mathbf{N}^{(\mathrm{s})}$ is randomly generated. By choosing $\mathbf{P}$, it is possible to make $\mathbf{N}^{(s)}$ generate any sublattice of $LAT(\mathbf{N}^{(d)})$.

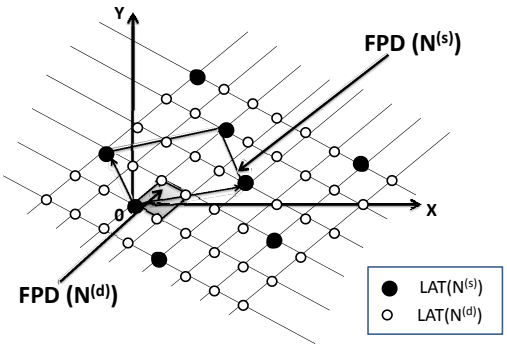


Figure 3.2: FPDs of dense lattice generator $\mathbf{N}^{(\mathrm{d})}$ and sparse lattice generator $\mathbf{N}^{(\mathrm{s})}$, where $\mathbf{N}^{(\mathrm{s})} = \mathbf{N}^{(\mathrm{d})}\mathbf{P}$. Here $\mathbf{N}^{(\mathrm{d})}$ is randomly generated (with the shaded FPD) and $\mathbf{P}$ is as given in the text. Also shown are the sensors belonging to the sparse and dense lattices where $LAT(\mathbf{N}^{(\mathrm{s})})$ is included within $LAT(\mathbf{N}^{(\mathrm{d})})$.

- Notice that $\det(\mathbf{N}^{(\mathrm{s})}) = \det(\mathbf{N}^{(\mathrm{d})})\det(\mathbf{P})$ where $\det(\mathbf{P}) > 1$ (we will not choose $\mathbf{P}$ to be unimodular). So the volume of the FPD of $\mathbf{N}^{(\mathrm{s})}$ is $\det(\mathbf{P}) > 1$ times larger than that of $\mathbf{N}^{(\mathrm{d})}$.

Lemma 3.5.1. *The cross difference co-array between $LAT(\mathbf{N}^{(\mathrm{s})})$ and $LAT(\mathbf{N}^{(\mathrm{d})})$ has all sensors located on the dense lattice generated by $\mathbf{N}^{(\mathrm{d})}$.* □

Proof. The cross difference co-array between $LAT(\mathbf{N}^{(\mathrm{s})})$ and $LAT(\mathbf{N}^{(\mathrm{d})})$ has sensors located at $\mathbf{N}^{(\mathrm{s})}\mathbf{n}^{(\mathrm{s})} - \mathbf{N}^{(\mathrm{d})}\mathbf{n}^{(\mathrm{d})}$ where $\mathbf{n}^{(\mathrm{s})}, \mathbf{n}^{(\mathrm{d})}$ are integer vectors. Since $\mathbf{N}^{(\mathrm{s})} = \mathbf{N}^{(\mathrm{d})}\mathbf{P}$, the differences can

be rewritten as $\mathbf{N}^{(\mathrm{d})}(\mathbf{P}\mathbf{n}^{(\mathrm{s})} - \mathbf{n}^{(\mathrm{d})}) = \mathbf{N}^{(\mathrm{d})}\acute{\mathbf{n}}$ where $\acute{\mathbf{n}}$ is an integer vector. Hence, the cross difference co-array has its sensors on the dense lattice $\mathbf{N}^{(\mathrm{d})}$. □

In practice, when the indices $\mathbf{n}^{(\mathrm{s})}$ and $\mathbf{n}^{(\mathrm{d})}$ take up a finite number of values, they need to be suitably chosen so that the resulting cross difference co-array describes a $2D$ array on $LAT(\mathbf{N}^{(\mathrm{d})})$ without any holes (missing sensors) inside (see Definition 3.3).

An array which is a union of two arrays: one with sensors given by $\mathbf{N}^{(\mathrm{s})}\mathbf{n}^{(\mathrm{s})}$ and the other with sensors given by $\mathbf{N}^{(\mathrm{d})}\mathbf{n}^{(\mathrm{d})}$, where $\mathbf{n}^{(\mathrm{s})}$ and $\mathbf{n}^{(\mathrm{d})}$ are integer vectors taking up a finite number of values, will be generally referred to as a nested $2D$ array. A more precise definition of the $2D$ nested array describing the exact location of sensors on the lattice is given in Definition 3.4. The term "nested" essentially indicates the fact that one lattice is contained within the other lattice. The $2D$ nested array reduces to the $1D$ nested array [135] when the matrix $\mathbf{N}^{(d)}$ is singular with identical columns and $\mathbf{P}$ is a diagonal matrix with equal elements. We will now discuss some of the key properties of this nested array which will also tell us how to choose the integer vectors $\mathbf{n}^{(\mathrm{s})}$ and $\mathbf{n}^{(\mathrm{d})}$.

3.5.1 Properties of the Nested Array

We are interested in the following question regarding the $2D$ nested array: given a sparse lattice generator $\mathbf{N}^{(\mathrm{s})}$ and a dense lattice generator $\mathbf{N}^{(\mathrm{d})}$, how to place $N^{(\mathrm{s})}$ sensors on the former and $N^{(\mathrm{d})}$ sensors on the latter so that the resulting difference co-array has $N^{(\mathrm{s})}N^{(\mathrm{d})}$ sensors on lattice $\mathbf{N}^{(\mathrm{d})}$ without any holes in the interior? The following theorem addresses this question. To state the theorem, we first define

$$SFPD(\mathbf{N}^{(\mathrm{s})}, k_1, k_2) \triangleq \left\{\mathbf{N}^{(\mathrm{s})}\Big([k_1 \quad k_2]^T - \mathbf{x}\Big), \mathbf{x} \in [0,1)^2\right\}$$

Here k_1, k_2 can be arbitrary integers and $SFPD$ essentially refers to "shifted FPD", where $FPD(\mathbf{N}^{(s)})$ is shifted by the vector $[k_1 \quad k_2]^T$.

Theorem 3.5.1. *Consider two non singular 2×2 matrices $\mathbf{N}^{(\mathrm{s})}$ and $\mathbf{N}^{(\mathrm{d})}$ related by an integer matrix $\mathbf{P}$ as $\mathbf{N}^{(\mathrm{s})} = \mathbf{N}^{(\mathrm{d})}\mathbf{P}$. Then*

1. *Any point $\mathbf{N}^{(\mathrm{d})}\mathbf{n}$ on $LAT(\mathbf{N}^{(\mathrm{d})})$ can be expressed as $\mathbf{N}^{(\mathrm{d})}\mathbf{n} = \mathbf{N}^{(\mathrm{s})}\mathbf{n}^{(\mathrm{s})} - \mathbf{N}^{(\mathrm{d})}\mathbf{n}^{(\mathrm{d})}$ where $\mathbf{n}^{(\mathrm{s})}$ is an integer vector and $\mathbf{n}^{(\mathrm{d})} \in FPD(\mathbf{P})$.*

2. *All points on $LAT(\mathbf{N}^{(\mathrm{d})})$ contained within $SFPD(\mathbf{N}^{(\mathrm{s})}, k_1, k_2)$ can be generated by the differences $\{\mathbf{N}^{(\mathrm{s})}[k_1 \quad k_2]^T - \mathbf{N}^{(\mathrm{d})}\mathbf{n}^{(\mathrm{d})}, \mathbf{n}^{(\mathrm{d})} \in FPD(\mathbf{P})\}$.*

♠ □

Proof. 1. Since $\mathbf{P}$ is an integer matrix, by the integer division theorem [182] we can say that given any integer vector $\mathbf{n}$, there exist unique integer vectors $\mathbf{n}^{(\mathrm{s})}$ and $\mathbf{n}^{(\mathrm{d})} \in FPD(\mathbf{P})$ such that

$$\mathbf{n} = \mathbf{P}\mathbf{n}^{(\mathrm{s})} - \mathbf{n}^{(\mathrm{d})}. \tag{3.3}$$

Hence, if $\mathbf{n}^{(\mathrm{s})}$ is allowed to take up all possible integer values and $\mathbf{n}^{(\mathrm{d})}$ is allowed to take up all integer values in $FPD(\mathbf{P})$, then all integer vectors $\mathbf{n}$ can be generated. Premultiplying (3.3) by $\mathbf{N}^{(\mathrm{d})}$, we get the desired result.

2. Note that a point $\mathbf{N}^{(\mathrm{d})}\acute{\mathbf{n}}$ on $LAT(\mathbf{N}^{(\mathrm{d})})$ also lies in $SFPD(\mathbf{N}^{(\mathrm{d})}, k_1, k_2)$ iff

$$\mathbf{N}^{(\mathrm{d})}\acute{\mathbf{n}} = \mathbf{N}^{(\mathrm{s})}[k_1 \quad k_2]^T - \mathbf{N}^{(\mathrm{s})}\mathbf{x}, \quad \mathbf{x} \in [0 \quad 1)^2.$$

This is equivalent to $\acute{\mathbf{n}} = \mathbf{P}[k_1 \quad k_2]^T - \mathbf{n}^{(\mathrm{d})}$ where $\mathbf{n}^{(\mathrm{d})} = \mathbf{P}\mathbf{x}$. Hence $\mathbf{n}^{(\mathrm{d})}$ is an integer vector inside $FPD(\mathbf{P})$. This holds for each $\acute{\mathbf{n}}$ such that $\mathbf{N}^{(\mathrm{d})}\acute{\mathbf{n}}$ belongs to $SFPD(\mathbf{N}^{(\mathrm{s})}, k_1, k_2)$. Therefore all elements on $LAT(\mathbf{N}^{(\mathrm{d})})$ inside $SFPD(\mathbf{N}^{(\mathrm{s})}, k_1, k_2)$ can be generated by the set of differences $\{\mathbf{N}^{(\mathrm{s})}[k_1 \quad k_2]^T - \mathbf{N}^{(\mathrm{d})}\mathbf{n}^{(\mathrm{d})}, \mathbf{n}^{(\mathrm{d})} \in FPD(\mathbf{P})\}$.

□

3.5.2 A Specific Construction

Let us consider $N^{(\mathrm{s})} = N_1^{(\mathrm{s})}N_2^{(\mathrm{s})}$ sensors on the sparse lattice $\mathbf{N}^{(\mathrm{s})}$ located at $\mathbf{N}^{(\mathrm{s})}[k_1 \quad k_2]^T, 0 \leq k_1 \leq N_1^{(\mathrm{s})} - 1, 0 \leq k_2 \leq N_2^{(\mathrm{s})} - 1$. Let us also place $N^{(\mathrm{d})} = \det(\mathbf{P})$ elements on a dense lattice $\mathbf{N}^{(\mathrm{d})}$ whose locations are given by $\mathbf{N}^{(\mathrm{d})}\mathbf{n}^{(\mathrm{d})}, \mathbf{n}^{(\mathrm{d})} \in FPD(\mathbf{P})$. Here $\mathbf{P}$ is given by $\mathbf{P} = \begin{pmatrix} 3 & 2 \\ 2 & 4 \end{pmatrix}$. Then $SFPD(\mathbf{N}^{(\mathrm{s})}, k_1, k_2)$ for each k_1, k_2 in the above mentioned range, is completely filled with sensors on the dense lattice as we consider the cross difference co-array between the sparse and dense arrays. This is pictorially depicted in Fig 3.3 where (a) shows the FPD of the dense lattice and the location of the integer vectors inside the FPD of the integer matrix $\mathbf{P}$ which decide the location of the det($\mathbf{P}$) dense lattice elements. Fig. 3.3 (b) shows how this choice of the dense lattice

sensors fills up $SFPD(\mathbf{N}^{(\mathrm{s})}, k_1, k_2)$ for $k_1 = 1, k_2 = 1$. In other words, each of the $N_1^{(\mathrm{s})} N_2^{(\mathrm{s})}$ $SFPD$s in the sparse array is completely filled with the $N^{(\mathrm{d})}$ =det($\mathbf{P}$) dense lattice points, yielding a total of det($\mathbf{P}$)$N_1^{(\mathrm{s})} N_2^{(\mathrm{s})}$ virtual sensors using only $N_1^{(\mathrm{s})} N_2^{(\mathrm{s})}$+det($\mathbf{P}$) physical sensors.

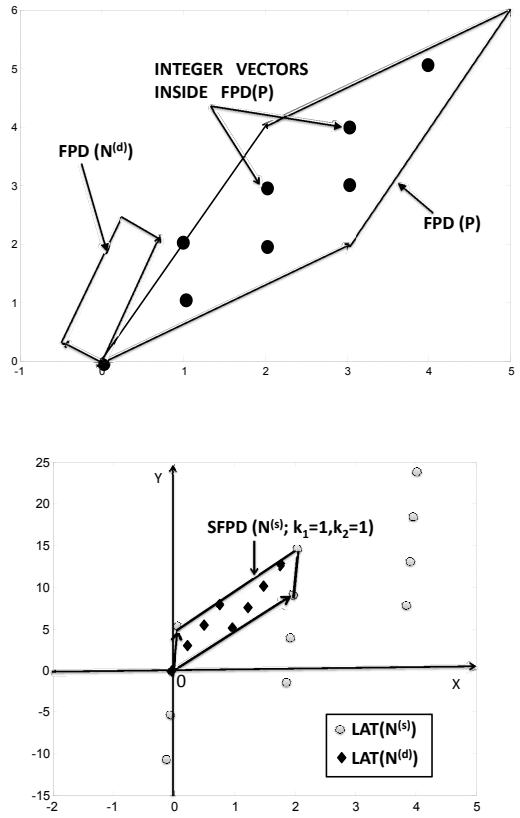


Figure 3.3: (top) The integer vectors inside $FPD(\mathbf{P})$ (total number of such vectors is det($\mathbf{P}$)) which denote the locations of the sensors on the dense lattice generated by $\mathbf{N}^{(\mathrm{d})}$ (here $\mathbf{N}^{(\mathrm{d})}$ is randomly generated and $\mathbf{P}$ is given in the text). Also shown is $FPD(\mathbf{N}^{(\mathrm{d})})$. (bottom) This choice of location of det($\mathbf{P}$) sensors on the dense lattice $LAT(\mathbf{N}^{(\mathrm{d})})$ completely tiles $SFPD(\mathbf{N}^{(\mathrm{s})}, k_1, k_2)$. In this example, $k_1 = 1, k_2 = 1$.

This leads us to the following configuration of $2D$ nested array:

Definition 3.4: (Two dimensional Nested Array: Configuration I): A two dimensional nested array described by a 2×2 non singular matrix $\mathbf{N}^{(\mathrm{d})}$, an integer matrix $\mathbf{P}$ and integers $N^{(\mathrm{s})}$, $N^{(\mathrm{d})}$=det($\mathbf{P}$), is said to be of configuration I if the physical sensors are chosen as the union of two 2 D arrays:

(a) A dense array with $N^{(\mathrm{d})}$=det($\mathbf{P}$) elements on lattice generated by $\mathbf{N}^{(\mathrm{d})}$, with sensor locations

given by $\{\mathbf{N}^{(\mathrm{d})}\mathbf{n}^{(\mathrm{d})}, \mathbf{n}^{(\mathrm{d})} \in FPD(\mathbf{P})\}$.

(b) A sparse array with $N^{(\mathrm{s})}$ elements on the lattice generated by $\mathbf{N}^{(\mathrm{s})} = \mathbf{N}^{(\mathrm{d})}\mathbf{P}$, with sensor locations given by $\mathbf{N}^{(\mathrm{s})}[k_1 \quad k_2]^T$ with $0 \leq k_1 \leq N_1^{(\mathrm{s})} - 1, 0 \leq k_2 \leq N_2^{(\mathrm{s})} - 1$ and $N_1^{(\mathrm{s})} N_2^{(\mathrm{s})} = N^{(\mathrm{s})}$.

□

This definition is inspired by the cross differences of the form $\mathbf{N}^{(\mathrm{s})}[k_1 \quad k_2]^T - \mathbf{N}^{(\mathrm{d})}\mathbf{n}^{(\mathrm{d})}$ which tiles the sparse array with the dense array. However, the difference co-array also includes differences of the form $\mathbf{N}^{(\mathrm{d})}\mathbf{n}^{(\mathrm{d})} - \mathbf{N}^{(\mathrm{s})}[k_1 \quad k_2]^T$ which can correspond to an entirely or partially new set of sensor locations. We will address this issue in greater detail in Sec 3.6 and 3.7 and provide other configurations of nested array (by redefining the sensor locations).

An Example: Fig. 3.4 (a) shows a 2 level nested array with a randomly generated $\mathbf{N}^{(\mathrm{d})}$ and $\mathbf{P} = \begin{pmatrix} 3 & 1 \\ 3 & 3 \end{pmatrix}$, hence the dense array has $N^{(\mathrm{d})}$ =det$(\mathbf{P}) = 6$ sensors. We assume $N^{(s)} = 12$ and hence the sparse array has 12 sensors located on $LAT(\mathbf{N}^{(\mathrm{s})} = \mathbf{N}^{(\mathrm{d})}\mathbf{P})$. The difference co-array elements given by $\mathbf{N}^{(\mathrm{s})}\mathbf{n}^{(\mathrm{s})} - \mathbf{N}^{(\mathrm{d})}\mathbf{n}^{(\mathrm{d})}$ are shown in Fig. 3.4 (b). Note that there are $12 \times 6 = 72$ virtual sensors completely filling up the 12 shifted FPDs of the sparse lattice.

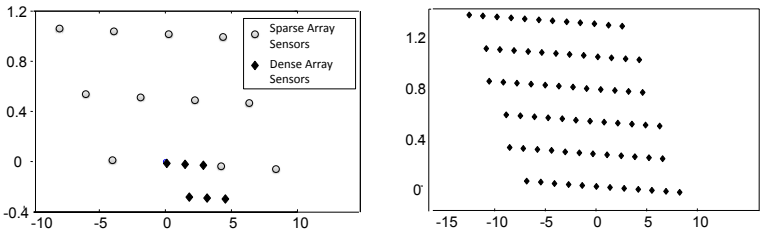


Figure 3.4: (left) A 2D nested array with 6 sensors on the dense lattice and 12 sensors on the sparse lattice, and (right) its corresponding difference co-array elements given by $\mathbf{N}^{(\mathrm{s})}\mathbf{n}^{(\mathrm{s})} - \mathbf{N}^{(\mathrm{d})}\mathbf{n}^{(\mathrm{d})}$. Here $\mathbf{N}^{(\mathrm{d})}$ is randomly generated and $\mathbf{P}$ is given in the text.

3.6 Orientation Issues of the Two Dimensional Nested Co-Array

In Sec. 3.5, we defined Configuration I for the $2D$ nested array, where a $2D$ virtual array with $N^{(\mathrm{d})}N^{(\mathrm{s})}$ sensors on $LAT(\mathbf{N}^{(\mathrm{d})})$ was obtained by considering the cross differences $\mathbf{N}^{(\mathrm{s})}\mathbf{n}^{(\mathrm{s})} - \mathbf{N}^{(\mathrm{d})}\mathbf{n}^{(\mathrm{d})}$

of $N^{(\mathrm{s})}$ sensors on the sparse lattice generated by $\mathbf{N}^{(\mathrm{s})}$ and $N^{(\mathrm{d})} = \det(\mathbf{P})$ suitably placed sensors on the dense lattice generated by $\mathbf{N}^{(\mathrm{d})}$. However, we can actually obtain even more sensors (about twice) on the virtual array by considering the negatives of these differences, viz., $\mathbf{N}^{(\mathrm{d})}\mathbf{n}^{(\mathrm{d})} - \mathbf{N}^{(\mathrm{s})}\mathbf{n}^{(\mathrm{s})}$. Henceforth, we will refer to the differences $\mathbf{N}^{(\mathrm{s})}\mathbf{n}^{(\mathrm{s})} - \mathbf{N}^{(\mathrm{d})}\mathbf{n}^{(\mathrm{d})}$ as the *positive half* and $\mathbf{N}^{(\mathrm{d})}\mathbf{n}^{(\mathrm{d})} - \mathbf{N}^{(\mathrm{s})}\mathbf{n}^{(\mathrm{s})}$ as the *negative half* of the $2D$ nested co-array.

A very interesting aspect of the $2D$ nested co-array is the orientation of these positive and negative halves in the $2D$ space. We illustrate this through the following three configurations of the $2D$ nested array and the relative orientations of the positive and negative halves that each gives rise to. The three configurations will be discussed under three cases:

1. Configuration I of the $2D$ nested array (see Definition 3.4) where the sensors in the union of the positive and negative halves are not contiguous.

2. Configuration II, where the positive and negative halves are contiguous and the overlap between them is reduced to a line.

3. Offset Configuration: It is a modification of Configuration II, obtained by placing the elements at an offset from their lattice positions. This configuration has the maximum number of contiguous virtual sensors among all the three configurations.

We will denote $\mathbf{N}^{(\mathrm{s})} = [\mathbf{n}_X^{(\mathrm{s})} \quad \mathbf{n}_Y^{(\mathrm{s})}]$ for the rest of the section.

Assumption: For better intuition into the symmetry related issues, for the rest of this section we shall assume that $\mathbf{P}$ is a *diagonal* matrix given by $\mathbf{P} = \begin{pmatrix} p_1 & 0 \\ 0 & p_2 \end{pmatrix}$. In the next section (Sec. 3.7), we shall show, using the Smith form decomposition, that this is not a loss of generality, because for every arbitrary (nonseparable) integer matrix $\mathbf{P}$, there is an equivalent representation in terms of a diagonal matrix.

3.6.1 Sensor Configuration I:

3.6.1.1 Sensor Locations

This configuration was described in Definition 3.4. Fig. 3.5 illustrates an example of this nested array and its difference co-array for the case $\mathbf{N}^{(\mathrm{d})} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \mathbf{P} = \begin{pmatrix} 3 & 0 \\ 0 & 3 \end{pmatrix}, N_1^{(\mathrm{s})} = 3, N_2^{(\mathrm{s})} = 6$.

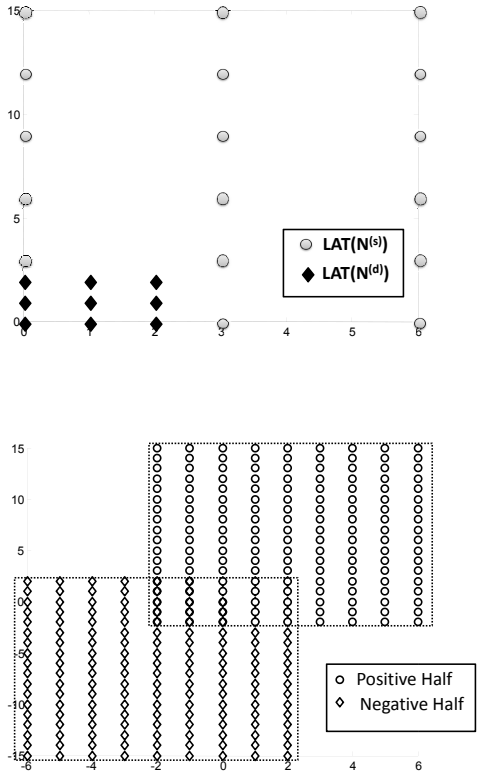


Figure 3.5: (top) A $2D$ nested array with sensor locations as given in Configuration I. (bottom) Corresponding difference co-array, where the positive and negative halves do not have contiguous sensors.

3.6.1.2 Positive and Negative Halves of the difference co-array

The positive half of the co-array is a "filled" $2D$ array in the sense that each $SFPD(\mathbf{N}^{(\mathrm{s})}, k_1, k_2)$ (see Sec 3.5) of the sparse array is completely filled by the dense array sensors for each k_1, k_2 in the sparse array. The elements of the positive half are given by $\{\ \mathrm{N}^{(\mathrm{d})}[n_1^{(\mathrm{diff})} \quad n_2^{(\mathrm{diff})}]^T, -(p_1-1) \leq n_1^{(\mathrm{diff})} < (N_1^{(\mathrm{s})}-1)p_1,$
$-(p_2-1) \leq n_2^{(\mathrm{diff})} < (N_2^{(\mathrm{s})}-1)p_2\}$, which satisfy the condition in Definition 3.3 for contiguous sensors. Similarly, the negative half, being the image of the positive half under point reflection about $\mathbf{0}$, is a filled array with contiguous sensors. However, the positive and negative halves together *do not give rise to contiguous sensors* on $LAT(\mathbf{N}^{(\mathrm{d})})$ (see Definition 3.3 for the mathematical meaning of contiguous sensors). This can be verified by taking the union of the positive half with its point reflection about $\mathbf{0}$. The positive half of the array is mostly in first quadrant, as a result of which the

negative half is mostly in third quadrant and hence their union do not describe contiguous sensors. Rather, they appear as two separate filled $2D$ arrays which share a band of common sensors around the sensor at $\mathbf{0}$ (see Fig.3.5 (b)). So, although we get almost twice as many virtual sensors by considering the negative half, they unfortunately are not contiguous. Ideally we would like the union of the positive and negative halves to describe contiguous sensors on $LAT(\mathbf{N}^{(\mathrm{d})})$, an example of which is provided in Fig 3.6. The following configuration provides a way to realize this.

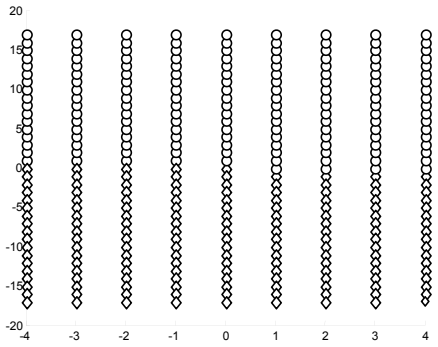


Figure 3.6: An Example of a $2D$ array with contiguous sensors.

3.6.2 Sensor Configuration II:

In this configuration, the sensors in the dense and sparse lattice are placed such that the resulting positive and negative halves of the difference co-array contain contiguous elements and they overlap only along a line. Also, the union of the two halves has more degrees of freedom than Configuration I. Here, we assume p_1 to be odd. The sensor locations and the resulting difference co-array are given by the following theorem:

Theorem 3.6.1. *Consider a $2D$ nested array where the sparse and dense array sensors are placed as follows:*

(a) The sparse array has $N^{(\mathrm{s})}$ sensors located at $\mathbf{N}^{(\mathrm{s})}[n_1^{(\mathrm{s})} \quad n_2^{(\mathrm{s})}]^T$ with $-N_1^{(\mathrm{s})} \leq n_1^{(\mathrm{s})} \leq N_1^{(\mathrm{s})}, 0 \leq n_2^{(\mathrm{s})} \leq N_2^{(\mathrm{s})} - 1, (2N_1^{(\mathrm{s})} + 1)N_2^{(\mathrm{s})} = N^{(\mathrm{s})}$.

(b) The dense array has $N^{(\mathrm{d})}$ sensors located at $\mathbf{N}^{(\mathrm{d})}\mathbf{n}^{(\mathrm{d})}$ where $\mathbf{n}^{(\mathrm{d})} = \mathbf{P}\mathbf{x}, \mathbf{x} \in [-0.5 \quad 0.5)\times(-1 \quad 0], N^{(\mathrm{d})} = \det(\mathbf{P})$.

The resulting positive and negative halves of the difference co-array overlap only along a line along $\mathbf{n}_X^{(\mathrm{s})}$.

There are $N_l = (2N_1^{(\mathrm{s})}+1)p_1$ sensors on the line of overlap. The union of the positive and negative halves of the co-array contains $2N^{(\mathrm{d})}N^{(\mathrm{s})} - N_l$ contiguous sensors on $LAT(\mathbf{N}^{(\mathrm{d})})$. ♠ □

Proof. The union of the positive and negative halves is now given by the virtual sensors located at
$\{\ \mathbf{N}^{(\mathrm{d})}[n_1^{(\mathrm{diff})} \quad n_2^{(\mathrm{diff})}]^T$,
$-N_1^{(\mathrm{s})}p_1 - (p_1-1)/2 \leq n_1^{(\mathrm{diff})} \leq N_1^{(\mathrm{s})}p_1 + (p_1-1)/2$,
$-N_2^{(\mathrm{s})}p_2 + 1 \leq n_2^{(\mathrm{diff})} \leq N_2^{(\mathrm{s})}p_2 - 1\}$ which describe contiguous sensors since the condition given by Definition 3.3 are satisfied. The positive half includes the sensors at

$$\{\mathbf{N}^{(\mathrm{d})}[k \quad 0]^T, -N_1^{(\mathrm{s})}p_1 - (p_1-1)/2 \leq k \leq N_1^{(\mathrm{s})}p_1 + (p_1-1)/2\}$$

and this line is identical to its own image under point reflection about $\mathbf{0}$. Hence both positive and negative halves share this common line of overlap. This line has $N_l = (2N_1^{(\mathrm{s})}+1)p_1$ sensors. Hence the union of the positive and negative halves has $2N^{(\mathrm{s})}N^{(\mathrm{d})} - N_l$ contiguous sensors on $LAT(\mathbf{N}^{(\mathrm{d})})$. □

So, the trick is to basically put the sparse and dense array sensors on *opposite* sides with respect to the origin as shown in Fig 3.7. It can be readily verified that the positive and negative halves *do not* overlap along a band but it is reduced to a line as demonstrated in Fig. 3.7. Here, $p_1 = 3$ and hence there are $3 \times 3 = 9$ sensors along the line of overlap as shown.

The value of p_1 is solely decided by the integer matrix $\mathbf{P}$ and it can be reduced by choosing $\mathbf{P}$ appropriately. We will visit this issue in Sec. 3.7.

3.6.3 Configuration on Offset-Lattice

As aptly pointed out by an anonymous reviewer, the line of overlap in the above configuration can be readily avoided by placing the sensors on the dense array at an offset from their positions on $LAT(\mathbf{N}^{(\mathrm{d})})$. To see this, let the sparse sensor positions be as in Configuration II, but let the elements on the dense array be located as $\mathbf{N}^{(\mathrm{d})}\mathbf{P}\mathbf{x}$ where $\mathbf{x} \in [-0.5 \quad 0.5) \times (-3/2 \quad -1/2]$. For the case where $\mathbf{N}^{(\mathrm{d})}$ is the identity matrix, this amounts to shifting the dense array (from Configuration II) along the $Y-$ axis by $1/2$ unit below the $y = 0$ line. This is illustrated in Fig. 3.8. It can be readily verified that the resulting difference co-array has *no line of overlap* between positive and negative halves. In fact, there are $2(2N_1^{(\mathrm{s})}+1)N_2^{(\mathrm{s})}p_1p_2$ elements given by the locations $\{\pm\mathbf{N}^{(\mathrm{d})}[n_1^{(\mathrm{diff})} \quad n_2^{(\mathrm{diff})}]^T$,
$-N_1^{(\mathrm{s})}p_1 - (p_1-1)/2 \leq n_1^{(\mathrm{diff})} \leq N_1^{(\mathrm{s})}p_1 + (p_1-1)/2$,

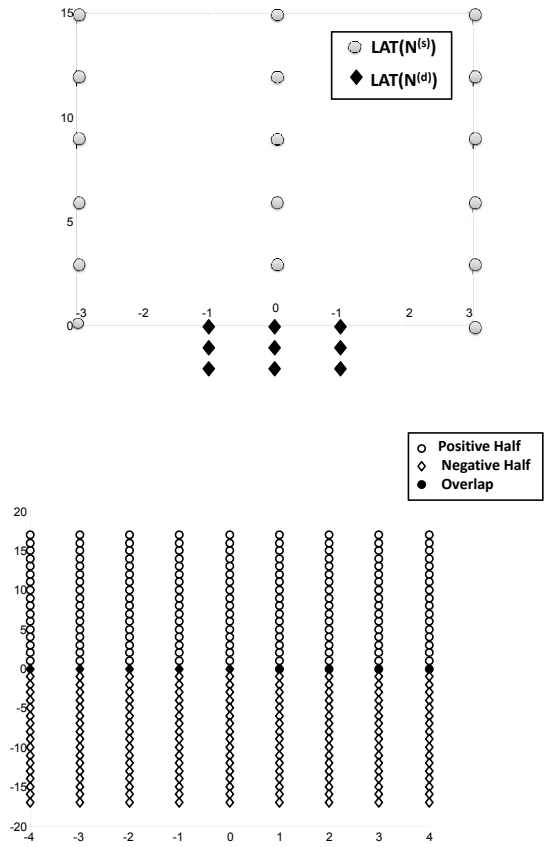


Figure 3.7: (top) A $2D$ nested array with sensor locations as given in Configuration II. (bottom) Corresponding difference co-array, where the positive and negative halves form a continuum and also their overlap is reduced to a line with only 9 sensors.

$1/2 \leq n_2^{(\text{diff})} \leq N_2^{(\text{s})} p_2 - 1/2\}$. Clearly offset configuration gives rise to more elements in the virtual array. However, it is to be noted:

1. The physical sensors in dense array, as well as the difference co-array are no more on the dense lattice. Infact they have been offset from their lattice positions, hence the name "Offset-lattice" configuration.

2. The physical sensor at **0** needs to be present in this offset-configuration, otherwise there will be holes in the difference co-array, a fact which is illustrated in Sec 3.7.3.

In the remaining chapter and also in accompanying Part II, we will treat both Configuration II (which retains the sensors on the lattice) and the "Offset" configuration and the reference will be made explicit.

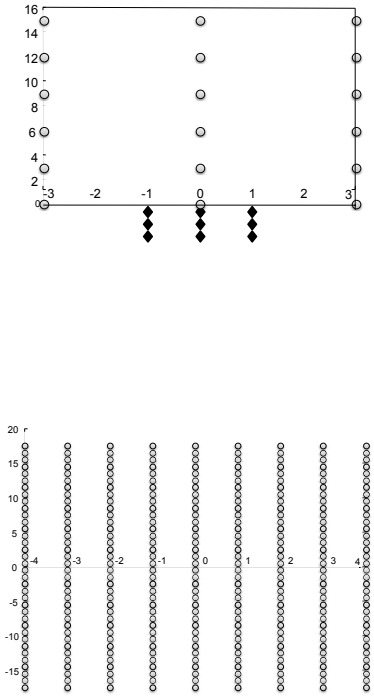


Figure 3.8: (top) A $2D$ nested array with sensor locations as given by Offset Configuration. (bottom) Corresponding difference co-array, with no overlap.

3.7 A Smith-Form Perspective to 2D Nested Array

In the proposed nested $2D$ array, the dense lattice generator $\mathbf{N}^{(\mathrm{d})}$ and the integer matrix $\mathbf{P}$ can be any arbitrary (non separable) integer matrix. This implies, in particular, the following:

1. The sparse and the dense lattice geometries can be chosen to be completely different. For example, the dense lattice FPD and the sparse lattice FPD might look very different from each other. This is illustrated in Fig. 3.9 where $\mathbf{N}^{(\mathrm{d})} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ and the integer matrix is $\mathbf{P} = \begin{pmatrix} 3 & 0 \\ -6 & 3 \end{pmatrix}$. Only in the case where $\mathbf{P}$ is a diagonal matrix, the sparse lattice FPD looks like a *stretched* version of the dense lattice FPD and their geometries look very similar. In general when $\mathbf{P}$ is non separable, the FPDs look quite different, and it might not even be

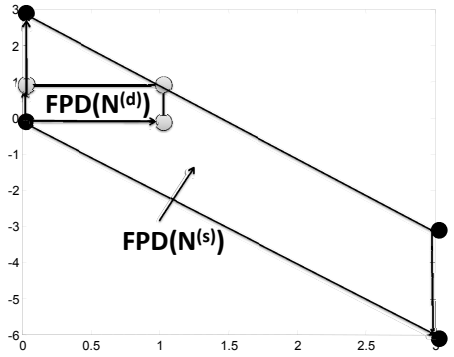


Figure 3.9: The FPD of $\mathbf{N}^{(s)}$ and $\mathbf{N}^{(d)}$ for the case where $\mathbf{P}$ is a non-diagonal integer matrix. Their geometries look very different.

obvious that one can be used to tile the other. Yet, by suitable sensor placement (as described earlier), the difference co-array can be constructed by completely filling up the sparse FPDs by the dense ones.

2. The virtual sensors in the difference co-array may not be contiguous as described in (3.1). This happens because $\mathbf{P}$ is any arbitrary integer matrix and not necessarily diagonal. However, for the $2D$ DOA estimation algorithm, we will need to ensure that the sensor locations in the difference co-array are indeed contiguous.

In this section, we shall address these two issues by viewing the co-array elements in terms of Smith form decomposition [182], which will allow us to construct a difference co-array where the elements are contiguous on the dense lattice even though $\mathbf{P}$ is an *arbitrary* integer lattice.

3.7.1 Smith Form of the integer matrix P

The Smith form decomposition of the integer matrix $\mathbf{P}$ is given by

$$\mathbf{P} = \mathbf{U_1 \Lambda U_2}$$

where $\mathbf{U_1}$ and $\mathbf{U_2}$ are integer unimodular matrices and $\mathbf{\Lambda} = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}$ is a diagonal integer matrix, and λ_1 divides λ_2. Then, the difference $\mathbf{N}^{(\mathrm{s})}\mathbf{n}^{(\mathrm{s})} - \mathbf{N}^{(\mathrm{d})}\mathbf{n}^{(\mathrm{d})}$ can be represented as $\mathbf{N}^{(\mathrm{s})}\mathbf{n}^{(\mathrm{s})} - \mathbf{N}^{(\mathrm{d})}\mathbf{n}^{(\mathrm{d})} = \mathbf{N}^{(\mathrm{d})}\mathbf{P}\mathbf{n}^{(\mathrm{s})} - \mathbf{N}^{(\mathrm{d})}\mathbf{n}^{(\mathrm{d})}$

$= \mathbf{N}^{(\mathrm{d})}\mathbf{U_1}\mathbf{\Lambda}\mathbf{U_2}\mathbf{n}^{(\mathrm{s})} - \mathbf{N}^{(\mathrm{d})}\mathbf{n}^{(\mathrm{d})}$

$= \mathbf{N}^{(\mathrm{d})}\mathbf{U_1}(\mathbf{\Lambda}\mathbf{U_2}\mathbf{n}^{(\mathrm{s})} - \mathbf{U_1}^{-1}\mathbf{n}^{(\mathrm{d})})$

$= \tilde{\mathbf{N}}^{(\mathrm{d})}(\mathbf{\Lambda}\tilde{\mathbf{n}}^{(\mathrm{s})} - \tilde{\mathbf{n}}^{(\mathrm{d})})$ where $\tilde{\mathbf{N}} = \mathbf{N}^{(\mathrm{d})}\mathbf{U_1}$ generates the same lattice as $\mathbf{N}^{(\mathrm{d})}$ since $\mathbf{U_1}$ is unimodular and

$$\tilde{\mathbf{n}}^{(\mathrm{s})} = \mathbf{U_2}\mathbf{n}^{(\mathrm{s})}, \quad \tilde{\mathbf{n}}^{(\mathrm{d})} = \mathbf{U_1}^{-1}\mathbf{n}^{(\mathrm{d})} \tag{3.4}$$

are integer vectors since $\mathbf{U_1}$ and $\mathbf{U_2}$ are unimodular. Hence, we can see that the difference co-array can be equivalently represented by the difference co-array of a dense generator $\tilde{\mathbf{N}}^{(\mathrm{d})}$ and a sparse lattice $\tilde{\mathbf{N}}^{(\mathrm{s})}$ where $\tilde{\mathbf{N}}^{(\mathrm{s})} = \tilde{\mathbf{N}}^{(\mathrm{d})}\mathbf{\Lambda}$ with sensor locations stipulated by $\tilde{\mathbf{n}}^{(\mathrm{s})}$ and $\tilde{\mathbf{n}}^{(\mathrm{d})}$ respectively. In particular, this implies the following:

1. The integer matrix $\mathbf{\Lambda}$ is a diagonal matrix with same determinant as $\mathbf{P}$. This implies that the lattice generators $\tilde{\mathbf{N}}^{(\mathrm{d})}$ and $\tilde{\mathbf{N}}^{(\mathrm{s})}$ have FPDs such that one is the stretched version of the other. This is illustrated in Fig.3.10. The physical sensors however, are located on the lattices generated by $\mathbf{N}^{(\mathrm{d})}$ and $\mathbf{N}^{(\mathrm{s})}$ whose FPDs appear quite different. Thus, for any arbitrary integer matrix $\mathbf{P}$, there is always an equivalent representation of the difference co-array in terms of another dense matrix $\tilde{\mathbf{N}}^{(\mathrm{d})}$ and sparse matrix $\tilde{\mathbf{N}}^{(\mathrm{s})}$ whose FPDs are similar in geometry.

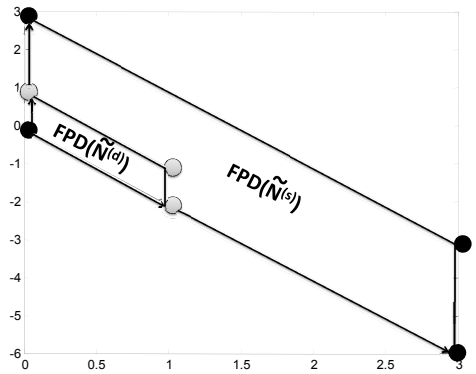


Figure 3.10: FPDs of $\tilde{\mathbf{N}}^{(s)}$ and $\tilde{\mathbf{N}}^{(d)}$ obtained by taking the Smith form of $\mathbf{P}$. The FPDs are scaled versions of each other with each side of $FPD(\tilde{\mathbf{N}}^{(\mathbf{s})})$ being 3 times that of $FPD(\tilde{\mathbf{N}}^{(d)})$ since $\mathbf{\Lambda} = 3\mathbf{I}$.

2. This will enable us to place the physical sensors in such a way that the difference co-array can be made to have contiguous sensors (as given by (3.1)) on $LAT(\tilde{\mathbf{N}}^{(\mathrm{d})})$ as described in the following subsection.

3.7.2 Co-array design in Smith domain

Here, we will consider $\tilde{\mathbf{N}}^{(\mathrm{d})}$ and $\tilde{\mathbf{N}}^{(\mathrm{s})}$ as our dense and sparse lattice generators respectively (which are related through the separable integer matrix $\mathbf{\Lambda}$) and select their respective index sets $\tilde{\mathbf{n}}^{(\mathrm{d})}$ and $\tilde{\mathbf{n}}^{(\mathrm{s})}$. The selection can be made according to Configuration II or Offset-configuration, both of which will be discussed. Denoting

$$\tilde{\mathbf{n}}^{(\mathrm{s})} = [\tilde{n}_1^{(\mathrm{s})} \quad \tilde{n}_2^{(\mathrm{s})}]^T, \tag{3.5}$$

we choose

$$-N_1^{(\mathrm{s})} \leq \tilde{n}_1^{(\mathrm{s})} \leq N_1^{(\mathrm{s})}, 0 \leq \tilde{n}_2^{(\mathrm{s})} < N_2^{(\mathrm{s})}. \tag{3.6}$$

The dense array sensors locations are given by $\tilde{\mathbf{N}}^{(\mathrm{d})}\tilde{\mathbf{n}}^{(\mathrm{d})}$ where

$$\tilde{\mathbf{n}}^{(\mathrm{d})} = \mathbf{\Lambda}\mathbf{x} \tag{3.7}$$

with

$$\mathbf{x} = [x_1 \quad x_2]^T, -0.5 \leq x_1 < 0.5, -1 + \alpha < x_2 \leq \alpha. \tag{3.8}$$

where

$$\alpha = \begin{cases} 0, & \text{Configuration II} \\ -0.5 & \text{Offset Configuration} \end{cases}$$

Also, we need the number of integer vectors in $FPD(\mathbf{\Lambda})$ along $[\lambda_1 \quad 0]^T$ to be odd, which implies λ_1 should be chosen to be odd (λ_1 corresponds to p_1). Hence, x_1 takes up values in the range

$$-(\lambda_1 - 1)/2 \leq x_1 \leq (\lambda_1 - 1)/2.$$

Choosing $\tilde{\mathbf{n}}^{(\mathrm{s})}$ and $\tilde{\mathbf{n}}^{(\mathrm{d})}$ to be in the above ranges, the actual physical sensor locations can be found by the transformation given by inverting (3.4). This selection of indices will ensure that the

positive half of the difference co-array has elements located at

$$\tilde{\mathbf{N}}^{(\mathrm{d})}[\tilde{n}_1^{(\mathrm{diff})} \quad \tilde{n}_2^{(\mathrm{diff})}]^T,$$

$$-N_1^{(\mathrm{s})}\lambda_1 - (\lambda_1 - 1)/2 \leq \tilde{n}_1^{(\mathrm{diff})} \leq N_1^{(\mathrm{s})}\lambda_1 + (\lambda_1 - 1)/2,$$

$$-\alpha \leq \tilde{n}_2^{(\mathrm{diff})} \leq (N_2^{(\mathrm{s})} - 1)\lambda_2 + \lambda_2 - 1 - \alpha$$

with a total of $(2N_1^{(\mathrm{s})} + 1)\lambda_1 N_2^{(\mathrm{s})}\lambda_2$ elements.

3.7.2.1 Smith form Based Design using Offset Configuration

It can be readily verified that the virtual array in this case has $2(2N_1^{(s)} + 1)N_2^{(s)}\lambda_1\lambda_2$ elements (with no loss of elements due to overlap), however they are *not* on $LAT(\mathbf{N}^{(d)})$, rather they are offset by half the inter element distance. The locations of co-array elements in the offset configuration are given by

$$\pm\left(\tilde{\mathbf{N}}^{(\mathrm{d})}[\tilde{n}_1^{(\mathrm{diff})} \quad \tilde{n}_2^{(\mathrm{diff})}]^T + \tilde{\mathbf{N}}^{(\mathrm{d})}[0 \quad 0.5]^T\right),$$

$$-N_1^{(\mathrm{s})}\lambda_1 - (\lambda_1 - 1)/2 \leq \tilde{n}_1^{(\mathrm{diff})} \leq N_1^{(\mathrm{s})}\lambda_1 + (\lambda_1 - 1)/2,$$

$$0 \leq \tilde{n}_2^{(\mathrm{diff})} \leq (N_2^{(\mathrm{s})} - 1)\lambda_2 + \lambda_2 - 1$$

3.7.2.2 Smith Form Based Design Using Configuration II

: For Configuration II, the positive and the negative halves overlap along a line (see Fig. 3.7) with N_l sensors where

$$N_l = (2N_1^{(\mathrm{s})} + 1)\lambda_1$$

Therefore, the positive and negative halves together produce a total of

$$2(2N_1^{(\mathrm{s})}+1)\lambda_1 N_2^{(\mathrm{s})}\lambda_2 - (2N_1^{(\mathrm{s})}+1)\lambda_1 \tag{3.9}$$

contiguous virtual sensors on $LAT(\tilde{\mathbf{N}}^{(\mathrm{d})})$ (i.e., $LAT(\mathbf{N}^{(\mathrm{d})})$). for any arbitrary integer matrix $\mathbf{P}$. This property will be particularly useful for the DOA estimation technique proposed in part II [143].

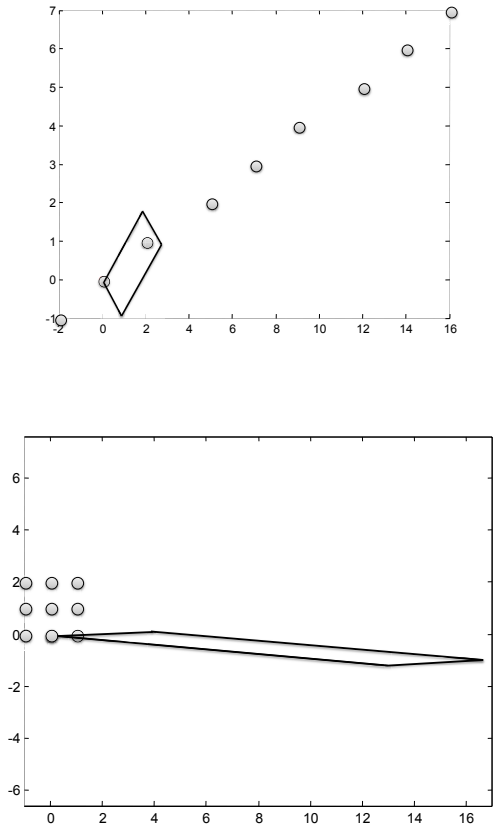


Figure 3.11: (Top) FPD of $\mathbf{N}^{(\mathrm{d})}$ and circles indicating the index set $\mathbf{n}_c^{(\mathrm{d})}$ (Bottom) FPD of $\tilde{\mathbf{N}}^{(\mathrm{d})} = \mathbf{N}^{(\mathrm{d})}\mathbf{U_1}$ and circles indicating the index set $\tilde{\mathbf{n}}_c^{(\mathrm{d})} = \mathbf{U_1}^{-1}\mathbf{n}_c^{(\mathrm{d})}$ which describe contiguous elements on $LAT(\tilde{\mathbf{N}}^{(\mathrm{d})})$.

The Smith form based design ensures that the co-array elements are contiguous on $LAT(\tilde{\mathbf{N}}^{(\mathrm{d})})$, i.e., they are given by $\tilde{\mathbf{N}}^{(\mathrm{d})}\tilde{\mathbf{n}}_c^{(\mathrm{d})}$ where the index set $\tilde{\mathbf{n}}_c^{(\mathrm{d})}$ spans a rectangular grid. The co-array sensors can be equivalently described by $\mathbf{N}^{(\mathrm{d})}\mathbf{n}_c^{(\mathrm{d})}$ where the index set $\mathbf{n}_c^{(\mathrm{d})} = \mathbf{U_1}\tilde{\mathbf{n}}_c^{(\mathrm{d})}$ however does not span a rectangular grid. The relation between the index sets $\mathbf{n}_c^{(\mathrm{d})}$ and $\tilde{\mathbf{n}}_c^{(\mathrm{d})}$ and the respective

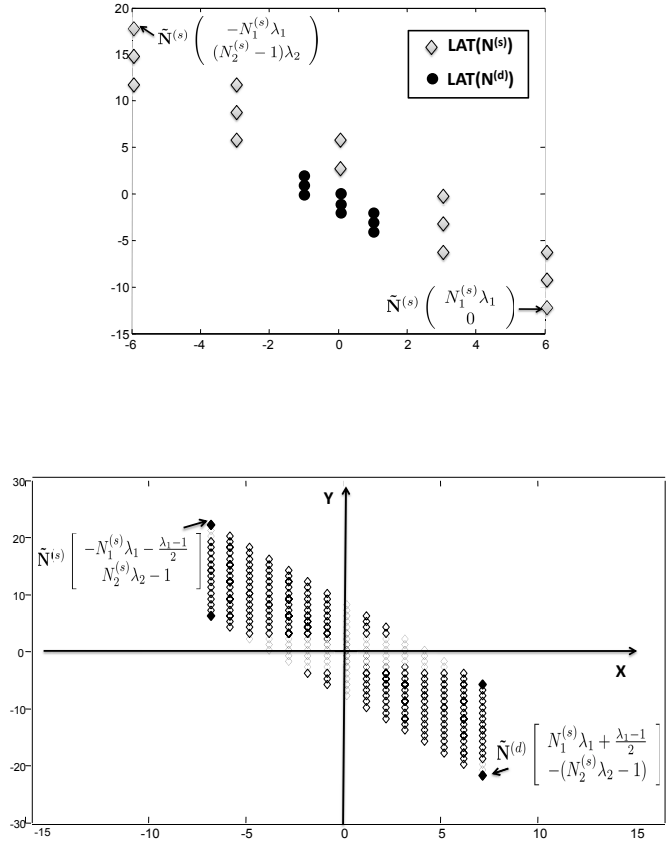


Figure 3.12: (Top) $2D$ Nested array designed in Smith domain showing the location of the sparse and dense sensors on $LAT(\tilde{\mathbf{N}}^{(\mathrm{s})})$ and $LAT(\tilde{\mathbf{N}}^{(\mathrm{d})})$ respectively with $\mathbf{\Lambda} = 3\mathbf{I}, N_1^{(\mathrm{s})} = 2, N_2^{(\mathrm{s})} = 3$. There are $(2N_1^{(\mathrm{s})}+1)N_2^{(\mathrm{s})} - 1 = 14$ sensors on the sparse and $\det(\mathbf{\Lambda}) = 9$ sensors on the dense lattice. (Bottom) Corresponding difference co-array with $2(2N_1^{(\mathrm{s})}+1)N_2^{(\mathrm{s})}\lambda_1\lambda_2 - (2N_1^{(\mathrm{s})}+1)\lambda_1 = 255$ sensors on $LAT(\tilde{\mathbf{N}}^{(\mathrm{d})})$.

FPDs of $\mathbf{N}^{(\mathrm{d})}$ and $\tilde{\mathbf{N}}^{(\mathrm{d})}$ are illustrated in Fig. 3.11 for $\mathbf{N}^{(\mathrm{d})} = \begin{pmatrix} 1 & 2 \\ -1 & 2 \end{pmatrix}$ and unimodular $\mathbf{U_1} = \begin{pmatrix} 2 & 7 \\ 1 & 3 \end{pmatrix}$.

The summary of the construction procedure for the nested array based on the Smith form, is given in Table 1. The total number of physical sensors thereby become

- $(2N_1^{(\mathrm{s})}+1)N_2^{(\mathrm{s})} - 1$ sensors on the sparse array
- $\lambda_1\lambda_2$ sensors on the dense lattice.

Table 3.1: **Summary of design of nested array in Smith domain.**

Given	A 2×2 nonsingular matrix (dense lattice generator) $\mathbf{N}^{(\mathrm{d})}$, an integer matrix $\mathbf{P}$ and integers $N_1^{(\mathrm{s})}, N_2^{(\mathrm{s})}$.
Step 1	Compute the Smith form of $\mathbf{P}$ as $\mathbf{P} = \mathbf{U}_1\mathbf{\Lambda}\mathbf{U}_2$, $\mathbf{\Lambda} = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}$.
Step 2	Construct the index set $\tilde{\mathbf{n}}^{(\mathrm{s})} = [\tilde{n}_1^{(\mathrm{s})} \quad \tilde{n}_2^{(\mathrm{s})}]^T$ as $-N_1^{(\mathrm{s})} \leq \tilde{n}_1^{(\mathrm{s})} \leq N_1^{(\mathrm{s})}, 0 \leq \tilde{n}_2^{(\mathrm{s})} \leq N_2^{(\mathrm{s})} - 1$,$\tilde{\mathbf{n}}^{(\mathrm{s})} \neq \mathbf{0}$.
Step 3	Construct the index set $\tilde{\mathbf{n}}^{(\mathrm{d})} = [\tilde{n}_1^{(\mathrm{d})} \quad \tilde{n}_2^{(\mathrm{d})}]^T$ as $-(\lambda_1 - 1)/2 \leq \tilde{n}_1^{(\mathrm{d})} \leq (\lambda_1 - 1)/2, \alpha \leq \tilde{n}_2^{(\mathrm{d})} \leq \lambda_2 - 1 + \alpha$ where $\alpha = -0.5$ for Offset configuration and 0 for Configuration II.
Step 4	Construct the transformed index sets $\mathbf{n}^{(\mathrm{s})}$ and $\mathbf{n}^{(\mathrm{d})}$ as $\mathbf{n}^{(\mathrm{s})} = \mathbf{U_2}^{-1}\tilde{\mathbf{n}}^{(\mathrm{s})}, \mathbf{n}^{(\mathrm{d})} = \mathbf{U_1}\tilde{\mathbf{n}}^{(\mathrm{d})}$.
Step 5	Place the sparse array elements at the locations given by $\mathbf{N}^{(\mathrm{d})}\mathbf{P}\mathbf{n}^{(\mathrm{s})}$ and the dense array at the locations $\mathbf{N}^{(\mathrm{d})}\mathbf{n}^{(\mathrm{d})}$.

Using these physical sensors, we can thus generate $2(2N_1^{(\mathrm{s})}+1)N_2^{(\mathrm{s})}\lambda_1\lambda_2 - (2N_1^{(\mathrm{s})}+1)\lambda_1$ contiguous sensors on $LAT(\tilde{\mathbf{N}}^{(\mathrm{d})})$. The following theory summarizes the Smith form design procedure for both configurations.

Theorem 3.7.1. *The difference co-array of a $2D$ nested array with the dense lattice generator $\mathbf{N}^{(\mathrm{d})}$ and sparse lattice generator $\mathbf{N}^{(\mathrm{s})}$ related through a nonseparable integer matrix $\mathbf{P}$ (i.e., $\mathbf{N}^{(\mathrm{s})} = \mathbf{N}^{(\mathrm{d})}\mathbf{P}$) can be equivalently represented as the difference co-array of a dense lattice generator $\tilde{\mathbf{N}}^{(\mathrm{d})}$ and sparse lattice generator $\tilde{\mathbf{N}}^{(\mathrm{s})}$ which are related through a separable integer matrix $\mathbf{\Lambda}$ ($\tilde{\mathbf{N}}^{(\mathrm{s})} = \tilde{\mathbf{N}}^{(\mathrm{d})}\mathbf{\Lambda}$). Furthermore if the nested array is designed following the procedure in Table I, then, for Configuration II, using a total of $(2N_1^{(\mathrm{s})}+1)N_2^{(\mathrm{s})} - 1$ sensors on the sparse array and $\lambda_1\lambda_2$ sensors on the dense array, a total of $2(2N_1^{(\mathrm{s})}+1)N_2^{(\mathrm{s})}\lambda_1\lambda_2 - (2N_1^{(\mathrm{s})}+1)\lambda_1$ contiguous sensors on $LAT(\tilde{\mathbf{N}}^{(\mathrm{d})})$ can be obtained in the difference co-array. For the Offset Configuration, $2(2N_1^{(\mathrm{s})}+1)N_2^{(\mathrm{s})}\lambda_1\lambda_2$ elements can be obtained in the difference co-array, which are at an offset from $LAT(\mathbf{N}^{(d)})$.* ♠ □

An example of the nested array in Smith domain is provided in Fig. 3.12. Following the design procedure in Smith domain, the physical array has 14 sensors on the sparse array and 8 sensors on the dense array. The resulting difference co-array has $2 \times (14+1) \times (8+1) - 15 = 255$ contiguous sensors on $LAT(\tilde{\mathbf{N}}^{(\mathrm{d})})$.

3.7.3 Inclusion of the 0 element

So far, we have included the zero sensor in the physical $2D$ nested array. We will now show that for Configuration II, if $N_1^{(\mathrm{s})} \geq 1$ and $N_2^{(\mathrm{s})}, \lambda_1, \lambda_2 > 1$, the same continuum of virtual sensors given by (3.9) can be achieved even when a physical sensor is not placed at the location $\mathbf{0}$. It will be demonstrated that this result is not true for the Offset Configuration.

Theorem 3.7.2. *Consider a $2D$ nested array designed in the Smith domain using Configuration II, as described in Sec 3.7.2. Let the sensor located at $\mathbf{0}$ be removed. Then, the virtual sensors thereby lost from the set of cross differences $\pm\{\tilde{\mathbf{N}}^{(\mathrm{s})}\tilde{\mathbf{n}}^{(\mathrm{s})} - \tilde{\mathbf{N}}^{(\mathrm{d})}\tilde{\mathbf{n}}^{(\mathrm{d})}\}$ can be regained by considering the self differences of the sparse and dense arrays individually, provided $N_1^{(\mathrm{s})} \geq 1$, $N_2^{(\mathrm{s})} > 1$, $\lambda_2 > 1$, and $\lambda_1 > 1$.* □

Proof. See Appendix 3.13.1. □

Comment on "Offset Configuration": The above theorem however does not hold if we place the dense and sparse array elements according to the "Offset configuration". Let us place the elements as in Fig 3.8 (a). with the only exception of the sensor at $\mathbf{0}$ being removed. The resulting difference co-array can be seen in Fig. 3.13 which clearly has holes.

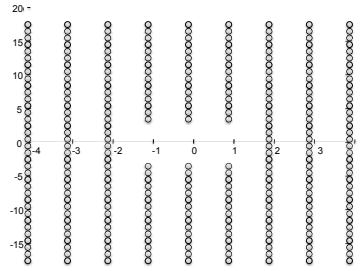


Figure 3.13: The co-array of the Offset Configuration without the physical element at position $\mathbf{0}$, resulting in holes.

3.8 Optimization of Degrees of Freedom

In this section, we will address the problem of maximizing the degrees of freedom (number of virtual sensors) in the difference co-array of a $2D$ nested array, given a fixed number of physical sensors.

3.8.1 Maximization of degrees of freedom in the offset configuration

We wish to find the distribution of the sensors between the sparse and the dense arrays that maximize the degrees of freedom. The total number of physical sensors in the $2D$ nested array in this case is $(2N_1^{(\mathrm{s})}+1)N_2^{(\mathrm{s})}-1+\lambda_1\lambda_2$, which is assumed to be constant ($=K$, say). In the Smith form of $\mathbf{P}$, λ_1 divides λ_2. However, we can always exchange λ_1 and λ_2 and permute the columns of $\mathbf{U}_1$ and the rows of $\mathbf{U}_2$ to yield the same integer matrix $\mathbf{P}$. So more generally, one of λ_1 and λ_2 should divide the other.

For the "Offset configuration", the optimization problem is simpler than that for Configuration II, and is given by

$$\max_{2N_1^{(\mathrm{s})}+1, N_2^{(\mathrm{s})}, \lambda_1, \lambda_2 \in \mathbb{N}^+} 2(2N_1^{(\mathrm{s})}+1)N_2^{(\mathrm{s})}\lambda_1\lambda_2 \tag{3.10}$$

subject to:

1. $(2N_1^{(\mathrm{s})}+1)N_2^{(\mathrm{s})}+\lambda_1\lambda_2 = K+1$.
2. λ_1 is odd.
3. One of λ_1 and λ_2 divides the other.

The solution to the above problem is given by integers $N_1^{(s)}$, $N_2^{(s)}$, λ_1 and λ_2 satisfying the values tabulated in Table 3.2. The result can be verified by using $AM-GM$ (Arithmetic Mean-Geometric Mean) inequality. As will be shown, due to the absence of a line of overlap, this produces more

Table 3.2: **Solution to Problem (3.10)**

K	odd	even
$(2N_1^{(\mathrm{s})}+1)N_2^{(s)}$	$(K+1)/2$	$(K+2)/2$
$\lambda_1\lambda_2$	$(K+1)/2$	$K/2$
DOF	$(K+1)^2/2$	$(K^2+2K)/2$

elements in virtual array than optimal value in Configuration II. However, the elements are no longer on a lattice, but at an offset.

3.8.2 Maximization of the degrees of freedom in Configuration II

For Configuration II, the physical sensor at **0** can be omitted provided

$$N_1^{(\mathrm{s})} \geq 1, \quad N_2^{(\mathrm{s})}, \lambda_1, \lambda_2 > 1. \tag{3.11}$$

We shall however assume that the physical sensors at **0** is present, otherwise we need to incorporate the extra constraints given by (3.11) which makes the integer optimization problem less tractable.

Ideally we would like to have a $2D$ nested array whose difference co-array is also a $2D$ array (i.e., it has sensors which are integer linear combinations of both the vectors $\tilde{\mathbf{n}}_1^{(\mathrm{d})}$ and $\tilde{\mathbf{n}}_2^{(\mathrm{d})}$), *not skewed towards either direction*. Defining

$$\Delta = N_2^{(\mathrm{s})}\lambda_2 - (2N_1^{(\mathrm{s})} + 1)\lambda_1,$$

it can be seen that positive Δ indicates how skewed the co-array is towards $\tilde{\mathbf{n}}_2^{(\mathrm{d})}$. The smaller the value of $|\Delta|$, the less skewed the array is, and when $|\Delta| = 0$, each half (positive and negative) of the co-array has same number of sensors along both $\tilde{\mathbf{n}}_1^{(\mathrm{d})}$ and $\tilde{\mathbf{n}}_2^{(\mathrm{d})}$. So, in order to avoid degenerate $1D$ solutions, one can fix Δ to a desired value and solve the optimization problem:

$$\max_{2N_1^{(\mathrm{s})}+1, N_2^{(\mathrm{s})}, \lambda_1, \lambda_2 \in \mathbb{N}^+} 2(2N_1^{(\mathrm{s})} + 1)N_2^{(\mathrm{s})}\lambda_1\lambda_2 - (2N_1^{(\mathrm{s})} + 1)\lambda_1 \tag{3.12}$$

subject to:

1. $(2N_1^{(\mathrm{s})} + 1)N_2^{(\mathrm{s})} + \lambda_1\lambda_2 = K + 1.$
2. λ_1 is odd.
3. One of λ_1 and λ_2 divides the other.
4. $N_2^{(\mathrm{s})}\lambda_2 - (2N_1^{(\mathrm{s})} + 1)\lambda_1 = \Delta.$

However, for a given Δ and K, the problem (3.12) may not have a feasible integer solution. In our

case, we would like to solve (3.12) for $\Delta = 0$, but for a given K, an integer solution may not exist. To resolve this, we will adopt the following strategy (henceforth referred to as strategy S_Δ):

1. Start with $\Delta = 0$ and try to find the optimal solution (if it exists) to (3.12).
2. If for some value of K, the solution does not exist, increase $|\Delta|$ in steps of 1 till we get a feasible solution and then solve (3.12) for that value of Δ.

Denoting $N_0 = (2N_1^{(\mathrm{s})} + 1)$, and defining

$$D_K \triangleq (K+1)^2/2 - (K+1)/2,$$

the solution obtained using strategy S_Δ is given by the following theorem.

Theorem 3.8.1. *The optimal degrees of freedom obtained by solving (3.12) following strategy S_Δ depends on values of K modulo 4 and are tabulated in Table II. The optimal values of $N_0, N_2^{(\mathrm{s})}, \lambda_1, \lambda_2$ that solve (3.12) are positive integers that satisfy the conditions tabulated in Table II along with the constraint that one of λ_1 and λ_2 divides the other.*

In each case, depending on K, several values of $N_1^{(\mathrm{s})}$, $N_2^{(\mathrm{s})}$, λ_1 and λ_2 can lead to the optimal value of DOF. Specific solutions that always exist for individual cases are:

1. $K+1 = 4p+2$: $\lambda_1 = 1, \lambda_2 = 2p+1, N_1^{(\mathrm{s})} = p, N_2^{(\mathrm{s})} = 1.$
2. $K+1 = 4p$: $\lambda_1 = 1, \lambda_2 = 2p+1, N_1^{(\mathrm{s})} = p-1, N_2^{(\mathrm{s})} = 1.$
3. $K+1 = 4p+1$: $\lambda_1 = 2p+1, \lambda_2 = 1, N_1^{(\mathrm{s})} = 0, N_2^{(\mathrm{s})} = 2p.$
4. $K+1 = 4p+3$: $\lambda_1 = 1, \lambda_2 = 2p+2, N_1^{(\mathrm{s})} = p, N_2^{(\mathrm{s})} = 1.$

□

Proof. The proof is elaborated in Appendix 3.13.2. □

Special Case of Degenerate Solution:

The last constraint in (3.12) is necessary to avoid solutions that degenerate to $1D$. Without it, the optimal degrees of freedom will be larger and it will correspond to $1D$ nested array. To see this, consider

$$\max_{2N_1^{(\mathrm{s})}+1, N_2^{(\mathrm{s})}, \lambda_1, \lambda_2 \in \mathbb{N}^+} 2(2N_1^{(\mathrm{s})}+1)N_2^{(\mathrm{s})}\lambda_1\lambda_2 - (2N_1^{(\mathrm{s})}+1)\lambda_1 \tag{3.13}$$

Table 3.3: **Solution to Problem (3.12) Using Strategy** S_Δ

$K+1$	$4p+2$	$4p$	$4p+1$	$4p+3$
Δ	0	2	-1	1
$N_0\lambda_1$	$2p+1$	$2p-1$	$2p+1$	$2p+1$
$N_2^{(\mathrm{s})}\lambda_2$	$2p+1$	$2p+1$	$2p$	$2p+2$
$N_0N_2^{(\mathrm{s})}$	$2p+1$	$2p-1$	$2p$	$2p+1$
$\lambda_1\lambda_2$	$2p+1$	$2p+1$	$2p+1$	$2p+2$
DOF	D_K	D_K-1	D_K-1	D_K

subject to:

1. $(2N_1^{(\mathrm{s})}+1)N_2^{(\mathrm{s})}+\lambda_1\lambda_2 = K+1$.

2. λ_1 is odd.

3. One of λ_1 and λ_2 divides the other.

Theorem 3.8.2. *In the optimal solution to (3.13), we always have* $N_1^{(\mathrm{s})}=0$, $\lambda_1=1$. *The optimal values of* $N_2^{(\mathrm{s})}$ *and* λ_2 *and the optimized degrees of freedom depend on whether* K *is even or odd and are given in Table 3.4.*

$\diamondsuit$ $\square$

Proof. See Appendix 3.13.3.

Table 3.4: **Solution to Problem (3.13)**

K	odd	even
$N_2^{(\mathrm{s})}$	$(K+1)/2$	$(K+2)/2$
λ_2	$(K+1)/2$	$K/2$
DOF	$(K+1)^2/2-1$	$(K+1)^2/2-3/2$

The difference co-array of the $2D$ nested array consists of elements on $LAT(\tilde{\mathbf{N}}^{(\mathrm{d})})$ where $\tilde{\mathbf{N}}^{(\mathrm{d})} = \mathbf{N}^{(\mathrm{d})}\mathbf{U_1}$. Denote

$$\tilde{\mathbf{N}}^{(\mathrm{d})} = [\tilde{\mathbf{n}}_1^{(\mathrm{d})} \quad \tilde{\mathbf{n}}_2^{(\mathrm{d})}]. \tag{3.14}$$

Since the optimal solution given by Theorem 3.8.2 always yields $N_1^{(\mathrm{s})}=0$, $\lambda_1=1$, it can be seen that the optimal $2D$ nested array actually corresponds to a *degenerate* $1D$ array oriented entirely along the vector $\tilde{\mathbf{n}}_2^{(\mathrm{d})}$. This is because *we did not explicitly impose a constraint to avoid this degeneracy*. This is precisely what is provided by the last constraint in (3.12) to avoid the degeneracy.

3.8.3 Optimal Solution with $1D$ Uniform Linear Arrays

For each of the four possible values of K modulo 4, it can be seen from Theorem 3.8.1 that there exists a solution which corresponds to the physical dense and sparse arrays being reduced to $1D$ manifolds (but the difference co-array is obviously $2D$) as illustrated in Fig. 3.14. This is given by the following corollary:

Corollary 3.8.1. *A particular solution to the optimization problem (3.13) always exists, that corresponds to two $1D$ uniform linear arrays (ULAs) oriented along the two columns of $\tilde{\mathbf{N}}^{(\mathrm{d})} = [\tilde{\mathbf{n}}_1^{(\mathrm{d})} \quad \tilde{\mathbf{n}}_2^{(\mathrm{d})}]$ with K_1 sensors (excluding the sensor at $\mathbf{0}$) in the ULA along $\tilde{\mathbf{n}}_1^{(\mathrm{d})}$ and K_2 sensors (including the sensor at $\mathbf{0}$) in the ULA along $\tilde{\mathbf{n}}_2^{(\mathrm{d})}$ where K_1 and K_2 depend on K as follows:*

- $K+1 = 4p+2$*:* $K_1 = 2p, K_2 = 2p+1$,
- $K+1 = 4p$*:* $K_1 = 2p-2$ *and* $K_2 = 2p+1$,
- $K+1 = 4p+1$*:* $K_1 = 2p$, $K_2 = 2p$, *and*
- $K+1 = 4p+3$*:* $K_1 = 2p$, $K_2 = 2p+2$.

$\diamondsuit$ $\square$

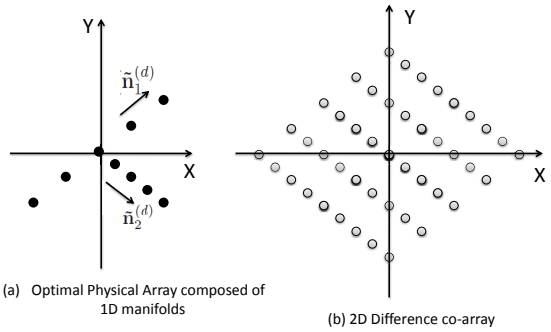


Figure 3.14: An example of the optimal solution where the physical array is union of two $1D$ ULAs, and the corresponding contiguous sensors on the difference co-array. Here $K = 9$, hence $K_1 = 4$, $K_2 = 5$. The total number of contiguous virtual elements in the difference co-array is $(K+1)^2/2 - (K+1)/2 = 45$.

It is to be noted that though the physical array is a combination of two ULAs, their difference co-array is still a $2D$ array and not a degenerate $1D$ array, unlike the solution given by Theorem 3.8.2.

3.9 Application of Nested Co-Array in 2D DOA estimation of more uncorrelated sources than sensors

In this section we will propose a novel algorithm for $2D$ DOA estimation by using the increased degrees of freedom offered by the co-array of the $2D$ nested array. The algorithm is capable of estimation of $2D$ DOAs of $O(MN)$ uncorrelated sources using only $O(M+N)$ physical sensors, which is not possible using traditional approaches. The key idea is to build a covariance-like matrix of much larger size ($O(MN \times MN)$) from the estimated covariance matrix (of size O($(M+N) \times (M+N)$) which

- is a positive semidefinite matrix even when the number of snapshots available is finite, and
- for large number of snapshots, behaves like the covariance matrix obtained from the (much larger) virtual difference co-array.

By applying conventional subspace based methods like MUSIC, one can estimate the $2D$ DOAs of $O(MN)$ sources. In the following subsections, we will develop a novel rank enhancing algorithm which serves this purpose. We will represent the received signal model (received by the physical array) by a model based on the larger difference co-array. By noticing the similarity of the model with that obtained in traditional array processing when sources are correlated, we draw inspiration from the $2D$ spatial smoothing technique [33] to build a suitable positive semidefinite matrix of appropriate rank. It is to be noted that this similarity is more of a qualitative nature, and quantitatively, there are technical differences from the actual correlated model. Also, our proposed algorithm differs considerably from the traditional spatial smoothing in many ways:

1. The proposed algorithm works in the co-array domain and hence leads to a covariance-like matrix of much larger dimension (corresponding to the difference co-array) which has the potential to resolve many more sources ($O(N^2)$ with only $O(N)$ physical sensors when subspace based algorithms are applied on it. However, in traditional array processing, the spatial smoothing technique works on the physical array (with N sensors) and it leads to a covariance matrix of size $(N/2) \times (N/2)$ which can resolve about half as many sources as the physical number of sensors.

2. The impinging signals are assumed uncorrelated in our proposed algorithm - they behave like

correlated sources (as explained later) when we go to the "co-array domain". However, traditionally spatial smoothing is used when the impinging sources are themselves correlated.

3. Traditionally in spatial smoothing, the signal and the noise are assumed uncorrelated; however, in our case, in the co-array domain, the signal and the noise terms become correlated and the cross terms between them cannot be ignored. This leads to a very different structure of the resulting matrix and the subsequent algebra becomes considerably different from that of conventional spatial smoothing.

3.9.1 Signal Model Based on the difference co-array:

Consider a $2D$ nested array with $(2N_1^{(\mathrm{s})}+1)N_2^{(\mathrm{s})}-1$ elements on the sparse lattice generated by $\mathbf{N}^{(\mathrm{s})}$ and det($\mathbf{P}$) elements on the dense lattice generated by $\mathbf{N}^{(\mathrm{d})}$, with $\mathbf{N}^{(\mathrm{s})}\mathbf{n}^{(\mathrm{s})}$ and $\mathbf{N}^{(\mathrm{d})}\mathbf{n}^{(\mathrm{d})}$ denoting the elements on these two lattices respectively.

Assume D narrowband signals impinging on this array with (θ_i, ϕ_i) being the azimuthal angle and elevation angle respectively and σ_i^2 being the power of the ith source, $i = 1, 2, \cdots, D$. The signal received at the output of these $(2N_1^{(\mathrm{s})}+1)N_2^{(\mathrm{s})}$+det($\mathbf{P}$) -1 sensors is given by

$$\mathbf{x}[k] = \begin{pmatrix} \mathbf{A_s} \\ \mathbf{A_d} \end{pmatrix} \mathbf{s}[k] + \boldsymbol{\eta}[k] \tag{3.15}$$

where $\mathbf{A_d}$ is a (det($\mathbf{P}$)) $\times D$ matrix and $\mathbf{A_s}$ is a $((2N_1^{(\mathrm{s})}+1)N_2^{(\mathrm{s})}-1) \times D$ matrix (corresponding to the dense and the sparse arrays respectively) whose elements are given by

$$[\mathbf{A_d}]_{i,k} = e^{j\frac{2\pi}{\lambda}\mathbf{u}_k^T\mathbf{N}^{(d)}\mathbf{n}_i^{(d)}},$$
$$k = 1, \cdots, D, i = 1, \cdots, \det(\mathbf{P})$$

$[\mathrm{A}_s]_{i,k} = e^{j\frac{2\pi}{\lambda}\mathbf{u}_k^T\mathbf{N}^{(s)}\mathbf{n}_i^{(s)}}$
$k = 1, \cdots, D, i = 1, \cdots, (2N_1^{(\mathrm{s})}+1)N_2^{(\mathrm{s})}-1$ where $\mathbf{u}_k = \sin\phi_k[\cos\theta_k \quad \sin\theta_k]^T$, and $\mathbf{s}[k] = [s_1[k] \cdots s_D[k]]^T$ denotes the source signal vector. The noise $\boldsymbol{\eta}[k]$ is assumed to be temporally and spatially white with power σ_n^2, and uncorrelated from the sources. We also assume the sources to be temporally uncorrelated so that the source autocorrelation matrix of $\mathbf{s}[k]$ (denoted by $\mathbf{R_{ss}}$) is diagonal with σ_i^2

being the ith diagonal element. Then,

$$\mathbf{R_{xx}} = E[\mathbf{x}\mathbf{x}^H] = \begin{pmatrix} \mathbf{A_s} \\ \mathbf{A_d} \end{pmatrix} \mathbf{R_{ss}} \begin{pmatrix} \mathbf{A_s} \\ \mathbf{A_d} \end{pmatrix}^H + \sigma_n^2 \mathbf{I} \tag{3.16}$$

Now, we vectorize $\mathbf{R_{xx}}$ to get

$$\mathbf{z} = \mathrm{vec}(\mathbf{R_{xx}}) = \begin{pmatrix} \mathbf{A_s} \\ \mathbf{A_d} \end{pmatrix}^* \odot \begin{pmatrix} \mathbf{A_s} \\ \mathbf{A_d} \end{pmatrix} \mathbf{p} + \sigma_n^2 \vec{\mathbf{1}}_n \tag{3.17}$$

where $\odot$ denotes the Khatri-Rao product (columnwise Kronecker product), $\mathbf{p}= [\sigma_1^2 \cdots \sigma_D^2]^T$ and $\vec{\mathbf{1}}_n= [\mathbf{e_1}^T \cdots \mathbf{e}^T_{(2N_1^{(\mathrm{s})}+1)N_2^{(\mathrm{s})}+\det(\mathbf{P})-1}]^T$ with $\mathbf{e}_i$ being a $((2N_1^{(\mathrm{s})}+1)N_2^{(\mathrm{s})}+\det(\mathbf{P})-1)\times 1$ column vector of all zeros except a 1 at the ith position. It can be verified that the rows of $\begin{pmatrix} \mathbf{A_s} \\ \mathbf{A_d} \end{pmatrix}^* \odot \begin{pmatrix} \mathbf{A_s} \\ \mathbf{A_d} \end{pmatrix}$ contain all rows $\mathbf{a}_{i,k}$ and $\mathbf{a}^*_{i,k}$ where $\mathrm{a}_{i,k} = [e^{j\frac{2\pi}{\lambda}\mathbf{u}_1^T(\mathbf{N}^{(\mathrm{s})}\mathbf{n}_i^{(\mathrm{s})}-\mathbf{N}^{(\mathrm{d})}\mathbf{n}_k^{(\mathrm{d})})} \cdots e^{j\frac{2\pi}{\lambda}\mathbf{u}_D^T(\mathbf{N}^{(\mathrm{s})}\mathbf{n}_i^{(\mathrm{s})}-\mathbf{N}^{(\mathrm{d})}\mathbf{n}_k^{(\mathrm{d})})}]$
$i = 1\cdots, (2N_1^{(\mathrm{s})}+1)N_2^{(\mathrm{s})}-1, \quad k = 1, \cdots \det(\mathbf{P})$ and also rows $\mathbf{a}^{(s)}_{i,k}, \mathbf{a}^{*(s)}_{i,k}, \mathbf{a}^{(d)}_{i,k}, \mathbf{a}^{*(d)}_{i,k}$ where $\mathrm{a}^{(s)}_{i,k} = [e^{j\frac{2\pi}{\lambda}\mathbf{u}_1^T(\mathbf{N}^{(\mathrm{s})}\mathbf{n}_i^{(\mathrm{s})}-\mathbf{N}^{(\mathrm{s})}\mathbf{n}_k^{(\mathrm{s})})} \cdots e^{j\frac{2\pi}{\lambda}\mathbf{u}_D^T(\mathbf{N}^{(\mathrm{s})}\mathbf{n}_i^{(\mathrm{s})}-\mathbf{N}^{(\mathrm{s})}\mathbf{n}_k^{(\mathrm{s})})}]$
$i,k = 1\cdots, (2N_1^{(s)}+1)N_2^{(s)}-1,\ \mathrm{a}^{(d)}_{i,k} = [e^{j\frac{2\pi}{\lambda}\mathbf{u}_1^T(\mathbf{N}^{(\mathrm{d})}\mathbf{n}_i^{(\mathrm{d})}-\mathbf{N}^{(\mathrm{d})}\mathbf{n}_k^{(\mathrm{d})})} \cdots e^{j\frac{2\pi}{\lambda}\mathbf{u}_D^T(\mathbf{N}^{(\mathrm{d})}\mathbf{n}_i^{(\mathrm{d})}-\mathbf{N}^{(\mathrm{d})}\mathbf{n}_k^{(d)})}]$
$i,k = 1\cdots, \det(\mathbf{P})$ These rows $\mathbf{a}_{i,k}, \mathbf{a}^{(d)}_{i,k}, \mathbf{a}^{(s)}_{i,k}$ and their conjugates, together behave like the array manifold of the difference co-array of the $2D$ nested array on which the D narrowband signals are impinging. Hence, we get $\{\mathrm{N}^{(\mathrm{s})}\mathbf{n}_i^{(\mathrm{s})} - \mathbf{N}^{(\mathrm{d})}\mathbf{n}_k^{(\mathrm{d})}, i = 1, \cdots, (2N_1^{(\mathrm{s})}+1)N_2^{(\mathrm{s})}-1,$
$k = 1, \cdots, \det(\mathbf{P})-1\}$
$\bigcup\{\mathbf{N}^{(\mathrm{s})}\mathbf{n}_i^{(\mathrm{s})} - \mathbf{N}^{(\mathrm{s})}\mathbf{n}_k^{(\mathrm{s})}, i,k = 1, \cdots, (2N_1^{(\mathrm{s})}+1)N_2^{(\mathrm{s})}-1\}$
$\bigcup\{\mathbf{N}^{(\mathrm{d})}\mathbf{n}_i^{(\mathrm{d})} - \mathbf{N}^{(\mathrm{d})}\mathbf{n}_k^{(\mathrm{d})}, i,k = 1, \cdots, \det(\mathbf{P})\}$
$= \{\tilde{\mathbf{N}}^{(\mathrm{d})}[k_1 \quad k_2]^T,$
$-N_1^{(\mathrm{s})}\lambda_1 - (\lambda_1-1)/2 \le k_1 \le \lambda_1 N_1^{(\mathrm{s})} + (\lambda_1-1)/2,$
$-((N_2^{(\mathrm{s})}-1)\lambda_2+\lambda_2-1) \le k_2 \le \lambda_2(N_2^{(\mathrm{s})}-1)+\lambda_2-1\}$ Hence the rows $\mathbf{a}_{i,k}, \mathbf{a}^*_{i,k}, \mathbf{a}^{(s)}_{i,k}, \mathbf{a}^{*(s)}_{i,k}, \mathbf{a}^{(d)}_{i,k}, \mathbf{a}^{*(d)}_{i,k}$ can be equivalently represented by the rows $\mathrm{b}_{k_1,k_2} = [e^{j\frac{2\pi}{\lambda}\mathbf{u}_1^T\tilde{\mathbf{N}}^{(\mathrm{d})}[k_1 \quad k_2]^T} \cdots e^{j\frac{2\pi}{\lambda}\mathbf{u}_D^T\tilde{\mathbf{N}}^{(d)}[k_1 \quad k_2]^T}]$
$-N_1^{(\mathrm{s})}\lambda_1 - (\lambda_1-1)/2 \le k_1 \le \lambda_1 N_1^{(\mathrm{s})} + (\lambda_1-1)/2,$
$-((N_2^{(\mathrm{s})}-1)\lambda_2 + \lambda_2 - 1) \le k_2 \le \lambda_2(N_2^{(\mathrm{s})}-1) + \lambda_2 - 1$

Let us construct the matrix $\mathbf{A}_{\mathrm{diff}}$ using these $2(2N_1^{(\mathrm{s})}+1)N_2^{(\mathrm{s})}\lambda_2\lambda_1 - (2N_1^{(\mathrm{s})}+1)\lambda_1$ rows. This amounts to extracting the corresponding rows from the vector $\mathbf{z}$ to get the $[2(2N_1^{(\mathrm{s})}+1)N_2^{(\mathrm{s})}\lambda_2\lambda_1 - (2N_1^{(\mathrm{s})}+1)\lambda_1]\times 1$ vector $\mathbf{z}_1$. This operation extracts one row with 1 (corresponding to the difference

of 0 in the difference co-array) and remaining "all-zero" rows from $\vec{\mathbf{1}}_n$, and we denote this vector as $\acute{\mathbf{e}}$, thereby obtaining

$$\mathbf{z}_1 = \mathbf{A}_{\text{diff}}\mathbf{p} + \sigma_n^2\acute{\mathbf{e}}. \tag{3.18}$$

This signal model behaves like the signal received by a "larger" array (given by the difference co-array) whose manifold is $\mathbf{A}_{\text{diff}}$. The source signal vector is replaced by the vector $\mathbf{p}$ of their powers (which behaves like fully correlated sources) and the noise being replaced by the deterministic vector $\acute{\mathbf{e}}$. Hence (3.18) can be thought of as being analogous to (3.15) through the corresponding analogies:

$$\mathbf{x}[k] \longrightarrow \mathbf{z}_1 \quad \begin{pmatrix} \mathbf{A_s} \\ \mathbf{A_d} \end{pmatrix} \longrightarrow \mathbf{A}_{\text{diff}}$$

$$\mathbf{s}[k] \longrightarrow \mathbf{p} \quad \eta[k] \longrightarrow \sigma_n^2\acute{\mathbf{e}}$$

In order to perform DOA estimation with this larger array manifold, we first need to build up a positive semidefinite matrix with appropriate rank on which subspace based algorithms (like MUSIC) can be applied. To achieve this, we propose a novel algorithm inspired from the idea of 2D spatial smoothing which will allow us to perform DOA estimation with the freedom of the larger difference co-array. The similarity of the co-array based signal model with that obtained from correlated sources impinging on the longer array enabled us to draw inspiration from spatial smoothing. The $1D$ versions of this algorithm for $1D$ nested arrays and coprime arrays were developed in [135] and [136] respectively. It is to be noted that we do not use spatial smoothing for decorrelating correlated sources, as it is done in the conventional sense [74],[163]. Also, in the conventional spatial smoothing, the noise and signal terms are uncorrelated and their cross terms can be averaged out to zero. In our case, however, the noise is also deterministic (behaving like fully correlated with signal) and the cross terms cannot be ignored.

3.9.2 Invariance in the Difference Co-Array

In order to apply spatial smoothing, the array is required to possess a certain invariance in its structure. The concept of invariance is heavily used in ESPRIT [157] based DOA estimation algorithms. In particular, invariance means that the array should be divisible into a number (say, K) of identical subarrays (of size $M_1 \times N_1$, say) which are merely shifted copies of each other. Let us consider a $2D$ array with contiguous sensors on $LAT(\tilde{\mathbf{N}}^{(\mathrm{d})})$ where the index set spans a rectangular grid of size $M \times N$. In the context of our $2D$ nested array,

$$M = (2N_1^{(\mathrm{s})} + 1)\lambda_1, N = 2N_2^{(\mathrm{s})}\lambda_2 - 1. \tag{3.19}$$

Also, let us consider a subarray of this array consisting of contiguous sensors on $LAT(\tilde{\mathbf{N}}^{(\mathrm{d})})$ where the index set spans a rectangular grid of size $M_1 \times N_1$ ($M_1 < M$, $N_1 < N$). The set of sensor locations of this subarray is given by $\{\mathbf{S}_{0,0}(i,j), 0 \leq i < M_1, 0 \leq j < N_1\}$. Now, to maximize the number of subarrays K, which can be obtained as shifted copies of this subarray, the subarrays should have maximum possible overlap. This is obtained by shifting the fundamental subarray elements on lattice of $\tilde{\mathbf{N}}^{(\mathrm{d})}$ by *successive integer vectors*. Mathematically, this means, the (m,n)th shifted subarray's sensor locations are given by

$$\mathbf{S}_{m,n}(i,j) = \mathbf{S}_{0,0}(i,j) + \tilde{\mathbf{N}}^{(\mathrm{d})}[m \quad n]^T, 0 \leq i < M_1, 0 \leq j < N_1$$

Fig. 3.15 illustrates this for $m = 6, n = 8$. Since $\mathbf{S}_{m,n}$ is also a subarray of the original array, this implies, in particular, that the original array should have sensors located at $\tilde{\mathbf{N}}^{(d)}\acute{\mathbf{n}}$ where $\acute{\mathbf{n}}$ spans a set of successive integer vectors. This is ensured by designing the array such that the co-array index set in the Smith domain spans a rectangular grid. Hence, the Smith form decomposition provides a direct way to ensure that the resultant array has the desired invariance. Also, the improvement in the rank is proportional to the number of subarrays (N_{sub}, say) [151],[33], whereas the degrees of freedom available after spatial smoothing is proportional to the size of the subarray $M_1 \times N_1$. Since the total number of elements in the array is constant, increasing the number of shifted copies (N_{sub}) implies that the fundamental subarray is smaller, and a larger fundamental subarray means we have smaller number of shifts N_{sub} available. So, the best strategy would be to make them equal which is achieved by selecting the subarray to be $(M+1)/2 \times (N+1)/2$. Note in our case, M and N are both odd (given by (3.19)) for the co-array of the $2D$ nested array. In that case, the shifts are

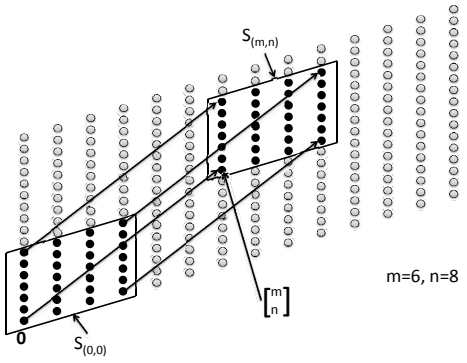


Figure 3.15: Subarrays of a $2D$ array with contiguous elements on a lattice. Here $\mathbf{S}_{0,0}$ is the fundamental subarray and $\mathbf{S}_{m,n}$ is the subarray shifted by the integer shift $[m \quad n]^T$ on the lattice, where $m = 6$, $n = 8$.

given by

$$\{[m \quad n]^T, 0 \leq m < (M+1)/2, 0 \leq n < (N+1)/2\}$$

and so $N_{\text{sub}} = \frac{(M+1)}{2}\frac{(N+1)}{2}$.

3.9.3 Novel Rank Enhancing Algorithm for DOA Estimation

To apply spatial smoothing, let us divide the co-array into $(M+1)(N+1)/4$ overlapping identical subarrays of size $(M+1)/2 \times (N+1)/2$. It is to be noted that this technique is a $2D$ generalization of the $1D$ method introduced in [135]. Here M, N are as defined in (3.19). The (m,n)th subarray $(m = 0, \cdots, (M+1)/2 - 1, \quad n = 0 \cdots, (N+1)/2 - 1)$ has sensors located at $\tilde{\mathbf{N}}^{(\mathrm{d})}\mathbf{v}_{m,n}$ where $\mathbf{v}_{m,n} = [v_{1m,n} \quad v_{2m,n}]^T$ such that $-(M-1)/2 + m \leq v_{1m,n} \leq m, \quad -(N-1)/2 + n \leq v_{2m,n} \leq n$. As shown in [135], this is equivalent to extracting the corresponding rows from $\mathbf{z}_1$ (by selecting the corresponding rows from $\mathbf{A}_{\text{diff}}$) to get the vector $\mathbf{z}_{m,n} = \mathbf{A}_{m,n}\mathbf{p} + \sigma_n^2\mathbf{e}_{m,n}$ where the elements of $\mathbf{A}_{m,n}$ are given as $[\mathrm{A}_{m,n}]_{\frac{(N+1)}{2}l+i,k} = e^{j\left(\omega_{1k}(l-(M-1)/2+m)+\omega_{2k}(i-(N-1)/2+n)\right)}$, $l = 0, \cdots, (M-1)/2, i = 0, \cdots, (N-1)/2$ Here,

$$\omega_{1,k} = \frac{2\pi}{\lambda}\sin\phi_k[\cos\theta_k \quad \sin\theta_k]\tilde{\mathbf{n}}_{\mathbf{1}}^{(\mathrm{d})}, \tag{3.20}$$

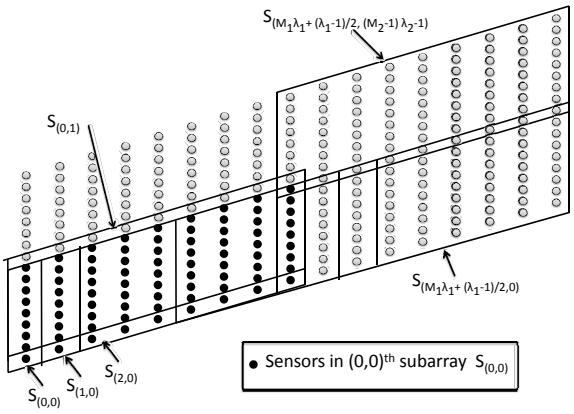


Figure 3.16: The virtual difference co-array array of $2D$ nested array and its subarrays used for the proposed DOA estimation algorithm based on spatial smoothing. The virtual array is of size $M \times N$ where $M = 17$, $N = 17$. The fundamental subarray is of size $(M+1)2 \times (N+1)/2 = 9 \times 9$ and there are a total of $(M+1)(N+1)/4 = 81$ subarrays, each of which is a shifted copy of the fundamental subarray.

$$\omega_{2,k} = \frac{2\pi}{\lambda} \sin\phi_k [\cos\theta_k \quad \sin\theta_k] \tilde{\mathbf{n}}_{\mathbf{2}}^{(\mathrm{d})} \tag{3.21}$$

where $\tilde{\mathbf{N}}^{(\mathrm{d})} = [\tilde{\mathbf{n}}_{\mathbf{1}}^{(\mathrm{d})} \quad \tilde{\mathbf{n}}_{\mathbf{2}}^{(\mathrm{d})}]$. The vector $\mathbf{e}_{m,n}$ is a $[(M+1)(N+1)/4] \times 1$ vector of all zeros except a 1 at the $((M+1)(N+1)/4 - \frac{(N+1)}{2}m - n)$th position. It can be readily verified that

$$\mathbf{z}_{m,n} = \mathbf{A}_{0,0} \mathbf{\Lambda}_1^m \mathbf{\Lambda}_2^n \mathbf{p} + \sigma_n^2 \mathbf{e}_{m,n}$$

where $\mathbf{\Lambda}_1$ and $\mathbf{\Lambda}_2$ are $D \times D$ diagonal matrices with (i,i)th element given by $e^{j\omega_{1i}}$ and $e^{j\omega_{2i}}$ respectively. We can now define $\mathrm{R}_{m,n} \triangleq \mathbf{z}_{m,n}\mathbf{z}_{m,n}^H$

$= \mathbf{A}_{0,0}\mathbf{\Lambda_1}^m\mathbf{\Lambda_2}^n\mathbf{p}\mathbf{p}^H(\mathbf{\Lambda_2}^n)^H(\mathbf{\Lambda_1}^m)^H\mathbf{A}_{0,0}^H + \sigma_n^4\mathbf{e}_{m,n}\mathbf{e}_{m,n}^H$

$+ \sigma_n^2\mathbf{A}_{0,0}\mathbf{\Lambda_1}^m\mathbf{\Lambda_2}^n\mathbf{p}\mathbf{e}_{m,n}^H + \sigma_n^2\mathbf{e}_{m,n}\mathbf{p}^H(\mathbf{\Lambda_1}^m\mathbf{\Lambda_2}^n)^H\mathbf{A}_{0,0}^H$ Taking the average of $\mathbf{R}_{m,n}$ over all (m,n), we get

$$\mathbf{R}_{\mathrm{co-array}} \triangleq \frac{1}{(M+1)(N+1)/4} \sum_{m=0}^{(M-1)/2} \sum_{n=0}^{(N-1)/2} \mathbf{R}_{m,n} \tag{3.22}$$

It can be shown that

$$\sum_{m=0}^{(M-1)/2} \sum_{n=0}^{(N-1)/2} \mathbf{\Lambda_1}^m\mathbf{\Lambda_2}^n\mathbf{p}\mathbf{p}^H(\mathbf{\Lambda_2}^n)^H(\mathbf{\Lambda_1}^m)^H = \mathbf{R}_{ss}\mathbf{A}_{0,0}^H\mathbf{A}_{0,0}\mathbf{R}_{ss}, \tag{3.23}$$

$$\sum_{m=0}^{(M-1)/2} \sum_{n=0}^{(N-1)/2} \mathbf{\Lambda_1}^m\mathbf{\Lambda_2}^n\mathbf{p}\mathbf{e}_{m,n}^H = \mathbf{R}_{ss}\mathbf{A}_{0,0}^H, \tag{3.24}$$

$$\sum_{m=0}^{(M-1)/2}\sum_{n=0}^{(N-1)/2}\mathbf{e}_{m,n}\mathbf{e}_{m,n}^{H}=\mathbf{I}_{(M+1)/2\times(N+1)/2}. \tag{3.25}$$

Substituting the values from (3.23), (3.24), (3.25) in (3.22), we get

$$\mathbf{R}_{\mathrm{co-array}}=(\mathbf{A}_{0,0}\mathbf{R}_{ss}\mathbf{A}_{0,0}^{H}+\sigma_n^2\mathbf{I}_{(M+1)/2\times(N+1)/2})^2. \tag{3.26}$$

We call the matrix $\mathbf{R}_{\mathrm{co-array}}$ as the rank-enhanced matrix and it enables us to perform DOA estimation with the freedom of $(N_1^{(\mathrm{s})}\lambda_1+(\lambda_1+1)/2)N_2^{(\mathrm{s})}\lambda_2$ $2D$ sensors using only $(2N_1^{(\mathrm{s})}+1)N_2^{(\mathrm{s})}+\lambda_1\lambda_2-1$ physical sensors, as given by the following theorem:

Theorem 3.9.1. *The matrix $\mathbf{R}_{\mathrm{co-array}}$ as defined in (3.22) can be expressed as $\mathbf{R}_{\mathrm{co-array}}=\hat{\mathbf{R}}^2$ where*

$$\hat{\mathbf{R}}=\frac{1}{\sqrt{\frac{(M+1)(N+1)}{4}}}(\mathbf{A}_{0,0}\mathbf{R}_{ss}\mathbf{A}_{0,0}^{H}+\sigma_n^2\mathbf{I}_{(M+1)/2\times(N+1)/2})$$

has the same form as the covariance matrix of the signal received by a (larger) $(M+1)/2\times(N+1)/2$ array with sensors on lattice $\tilde{\mathbf{N}}^{(\mathrm{d})}$ (whose manifold is represented by $\mathbf{A}_{0,0}\in\mathcal{C}^{\frac{(M+1)(N+1)}{4}\times D}$) and hence by applying MUSIC on $\mathbf{R}_{\mathrm{co-array}}$, two dimensional DOA estimation can be performed with the degree of freedom of a $(M+1)/2\times(N+1)/2$ sensor array with $(M+1)(N+1)/4=(N_1^{(s)}\lambda_1+(\lambda_1+1)/2)N_2^{(s)}\lambda_2$ elements. □

3.10 Identifiability Issues

In this section, we will explore the identifiability conditions for subspace based techniques applied to $2D$ DOA estimation using the nested array. The subspace based spectral estimation methods are applied on the covariance matrix $\mathbf{R}_{\mathrm{co-array}}$ which corresponds to the much larger co-array. Hence the identifiability problem is the same as that of traditional subspace based methods applied to a physical $2D$ array identical to the difference co-array in our case, the only difference being that the sensors are now virtually generated using far fewer number of physical sensors. Hence the results discussed in this section apply to a generic DOA estimation scenario using a $2D$ array with sensors uniformly placed on a lattice. The matrix $\mathbf{A}_{0,0}\mathbf{R}_{ss}\mathbf{A}_{0,0}^{H}$ in $\hat{\mathbf{R}}$ denotes the signal component of the covariance matrix and is usually rank deficient, since the number of sources D is less than the total number of sensors in the $2D$ array. However, in the case of $2D$ DOA estimation, the relation between the number of sources D and the rank of the signal subspace is not direct. In $1D$, the

array manifold of a ULA (which corresponds to $\mathbf{A_{0,0}}$ in $2D$) is a Vandermonde matrix and hence, whenever the D (number of sources) $< N$(number of sensors), the rank of the manifold is D, and hence the null spectrum (estimated using the $N - D$ null space eigenvectors) peaks correspond uniquely to the DOAs. However, in the case of $2D$ DOA estimation, the following situations may arise, *depending on how the source DOAs are located in* $2D$:

- The number of distinct sources is D but the column rank of the array manifold $\mathbf{A}_{0,0}$, is $< D$.
- The column rank of the array manifold $\mathbf{A}_{0,0}$ is equal to the number of sources D, but the steering vector corresponding to a different $2D$ DOA (which is not equal to any of the source DOAs), happens to lie in the range space of $\mathbf{A}_{0,0}$.

Both of these will lead to non unique identifiability of the source DOAs using any traditional subspace based method, even when $D < N_s$, N_s being the number of sensors under consideration. The issue of identifiability in $2D$ harmonic retrieval has been addressed previously in [113] and the references therein. However, they offer *statisical* bounds on the rank of the $2D$ array manifold and the number of uniquely identifiable sources (assuming the $2D$ DOA pairs are drawn uniformly at random) which might not hold true for a given DOA scenario. In this section, we attempt to explore the identifiability issue from a purely deterministic viewpoint and provide some sufficient conditions on the rank of the array manifold, and more importantly, on the identifiability of $2D$ DOAs.

3.10.1 Rank of the Array Manifold matrix in $2D$

To examine the rank of the array manifold, let us look into its algebraic structure. Redefining

$$M = N_1^{(\mathrm{s})}\lambda_1 + \frac{\lambda_1 + 1}{2}$$

(recall that λ_1 is odd), and

$$N = N_2^{(\mathrm{s})}\lambda_2$$

and denoting $\alpha_i = e^{-j\omega_{1i}}, \beta_i = e^{-j\omega_{2i}}$, the manifold of the $2D$ (co) array is given by

$$\mathbf{A}_{0,0}=\begin{pmatrix} 1 & 1 & 1 & 1 \\ \beta_1 & \beta_2 & \cdots & \beta_D \\ \vdots & \vdots & \vdots & \vdots \\ \beta_1^{(N-1)} & \beta_2^{(N-1)} & \cdots & \beta_D^{(N-1)} \\ \alpha_1 & \alpha_2 & \cdots & \alpha_D \\ \alpha_1\beta_1 & \alpha_2\beta_2 & \cdots & \alpha_D\beta_D \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_1\beta_1^{(N-1)} & \alpha_2\beta_2^{(N-1)} & \cdots & \alpha_D\beta_D^{(N-1)} \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_1^{(M-1)} & \alpha_2^{(M-1)} & \cdots & \alpha_D^{(M-1)} \\ \alpha_1^{(M-1)}\beta_1 & \alpha_2^{(M-1)}\beta_2 & \cdots & \alpha_D^{(M-1)}\beta_D \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_1^{(M-1)}\beta_1^{(N-1)} & \alpha_2^{(M-1)}\beta_2^{(N-1)} & \cdots & \alpha_D^{(M-1)}\beta_D^{(N-1)} \end{pmatrix} \tag{3.27}$$

Clearly this is not a Vandermonde matrix but it can be expressed as the Khatri Rao Product of two Vandermonde matrices $\mathbf{A}_x$ and $\mathbf{A}_y$, i.e.,

$$\mathbf{A}_{0,0} = \mathbf{A}_x \odot \mathbf{A}_y \tag{3.28}$$

$$\mathbf{A}_x = \begin{pmatrix} 1 & 1 & 1 & 1 \\ \alpha_1 & \alpha_2 & \cdots & \alpha_D \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_1^{(M-1)} & \alpha_2^{(M-1)} & \cdots & \alpha_D^{(M-1)} \end{pmatrix} \tag{3.29}$$

$$\mathbf{A}_y = \begin{pmatrix} 1 & 1 & 1 & 1 \\ \beta_1 & \beta_2 & \cdots & \beta_D \\ \vdots & \vdots & \vdots & \vdots \\ \beta_1^{(N-1)} & \beta_2^{(N-1)} & \cdots & \beta_D^{(N-1)} \end{pmatrix} \tag{3.30}$$

Even when the Vandermonde matrices $\mathbf{A}_x$ and $\mathbf{A}_y$ are full column rank, their Khatri-Rao product in general, *may not be full rank*, as we shall see in subsequent examples. In fact, there exists no

deterministic formula for the rank of the Khatri-Rao product of two full column rank Vandermonde matrices. However, statistical characterizations of the rank of the Khatri Rao product exists which ensures that it is full rank with probability one [113]. But, one might come up with possibly infinite number of cases where the D pairs of values of (ω_1, ω_2) can be so chosen that $\mathbf{A}_{0,0}$ is rank deficient. We will now provide a way to choose the D DOA pairs such that $\mathbf{A}_{0,0}$ is of full column rank. We will also describe examples which show the existence of difference classes of D values of the DOA pairs which always lead to $\mathbf{A}_{0,0}$ being rank deficient.

3.10.1.1 Sufficient Condition for full column rank of $\mathbf{A}_{0,0}$ for $D \leq MN$

In the following lemma, we provide a sufficient condition for making the matrix full column rank for each $D \leq MN$:

Lemma 3.10.1. *Let us consider MN distinct pairs of (ω_1, ω_2) given by*

$$(\omega_{1m}, \omega_{2n}), m = 1, 2, \cdots, M, n = 1, 2, \cdots, N.$$

Here $\{\omega_{1m}, 1 \leq m \leq M\}$ and $\{\omega_{2n}, 1 \leq n \leq N\}$ are arbitrary $M + N$ real numbers, each in the range $[-\pi \quad \pi)$ such that

$$\omega_{1m} \neq \omega_{1\acute{m}}, m \neq \acute{m}, \quad \omega_{2n} \neq \omega_{2\acute{n}}, n \neq \acute{n}.$$

If the D pairs of source DOAs ($D \leq MN$) are selected from these MN pairs, the matrix $\mathbf{A}_{0,0}$ will always have rank D. □

Proof. The MN pairs of DOAs give rise to MN steering vectors given by the MN columns of $\mathbf{A}_M^{(1)} \otimes \mathbf{A}_N^{(2)}$ where, for $(K = M, i = 1)$ and $(K = N, i = 2)$,

$$\mathbf{A}_K^{(i)} = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ e^{j\omega_{i1}} & e^{j\omega_{i2}} & \dots & e^{j\omega_{iK}} \\ \vdots & \vdots & \vdots & \vdots \\ e^{j(K-1)\omega_{i1}} & e^{j(K-1)\omega_{i2}} & \dots & e^{j(K-1)\omega_{iK}} \end{pmatrix}$$

Hence, $\mathbf{A}_M^{(1)}$ and $\mathbf{A}_N^{(2)}$ are respectively rank-M and rank-N Vandermonde matrices and hence the rank of their kronecker product is MN. Therefore these MN source DOA steering vectors are

linearly independent. So any $D \leq MN$ steering vectors selected from these MN vectors will also be linearly independent, thereby proving the lemma. □

An example of the MN DOA pairs is pictorially depicted in Fig. 3.10.1.1 for $M = 5, N = 6$. They lie at the possible intersections of M vertical lines (denoting M distinct values of ω_1) and N horizontal lines (denoting the N distinct values of ω_2).

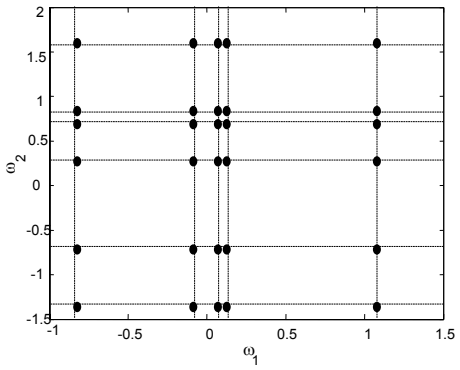


Figure 3.17: $MN = 30$ pairs of (ω_1, ω_2) obtained according to Lemma 3.10.1 by generating $M = 5$ random values of ω_1 and $N = 6$ random values of ω_2.

3.10.1.2 Column Rank Deficiency of the array manifold $\mathbf{A}_{0,0}$ when $D \leq MN$

There can be several ways in which the rank of $\mathbf{A}_{0,0}$ might be less than D even when there are $D \leq MN$ distinct $2D$ DOA pairs. We provide three classes of examples where this happens:

- *Example 1*: Assume $D = MN$. Let us define

$$P = \text{no. of distinct values in } \{\omega_{1i}, i = 1, 2, \cdots, D\}$$

$$Q = \text{no. of distinct values in } \{\omega_{2i}, i = 1, 2, \cdots, D\}$$

  Note that $P, Q \leq D$. When $P < M$ or $Q < N$, $\mathbf{A}_{0,0}$ is always column-rank deficient.

  Reason: Since $\mathbf{A}_{0,0}$ is the columnwise Kronecker product of $\mathbf{A}_x$ and $\mathbf{A}_y$, the columns of $\mathbf{A}_{0,0}$ is a subset of the columns of

$$\mathbf{A}_{\text{kron}} = \mathbf{A}_x \otimes \mathbf{A}_y.$$

Hence

$$\text{rank}(\mathbf{A}_{0,0}) \leq \text{rank}(\mathbf{A}_{\text{kron}}) = \text{rank}(\mathbf{A}_x)\text{rank}(\mathbf{A}_y) \tag{3.31}$$

When $P < M$, $\mathbf{A}_x$ is a Vandermonde matrix where only P out of D columns are distinct. Hence rank of $\mathbf{A}_x = P < M$. Similarly, rank of $\mathbf{A}_x = \min(Q, N)$. Hence when $P < M$ or $Q < N$, rank of $\mathbf{A}_{\text{kron}} = \min(P, M).\min(Q, N) < MN$. Thus, from (3.31), rank of $\mathbf{A}_{0,0} \leq \text{rank}(\mathbf{A}_{\text{kron}}) < MN$ thereby making $\mathbf{A}_{0,0}$ column-rank deficient.

- *Example 2*: Assuming $D > N$, if the D DOA pairs contain $N_1 > N$ distinct (α, β) pairs in (3.27) of the form

$$\{(\alpha_1, \beta_k), k = 1, 2, \cdots N_1\}, \tag{3.32}$$

then $\mathbf{A}_{0,0}$ has rank $< D$.

Reason: In this case, the N_1 pairs of DOAs give rise to the N_1 columns of the array manifold of the form

$$\begin{pmatrix} 1 \\ \alpha_1 \\ \vdots \\ \alpha_1^{(M-1)} \end{pmatrix} \otimes \underbrace{\begin{pmatrix} 1 & 1 & 1 & 1 \\ \beta_1 & \beta_2 & \cdots & \beta_{N_1} \\ \vdots & \vdots & \vdots & \vdots \\ \beta_1^{(N-1)} & \beta_2^{(N-1)} & \cdots & \beta_{N_1}^{(N-1)} \end{pmatrix}}_{\mathbf{A}_{N_1}}$$

Since $N_1 > N$, $\mathbf{A}_{N_1}$ is a Vandermonde matrix of rank N. Hence, these $N_1 > N$ columns are linearly dependent, and since $\mathbf{A}_{0,0}$ includes these N_1 columns, its columns are also linearly dependent, resulting in rank($\mathbf{A}_{0,0}$)$< D$.

- *Example 3*: Assume the D DOA pairs, denoted (α_i, β_i), $1 \leq i \leq D$, are given by

$$\alpha_i = \beta_i^k, 1 \leq i \leq D$$

where k is a given integer such that $k \in \mathbb{N}^+, 1 \leq k \leq N$. This implies that the D (ω_1, ω_2) pairs lie on a straight line through origin with integer slope k. Then, rank($\mathbf{A}_{0,0}$) $= \min(D, N +$

$(M-1)k)$.

Reason: In this case, the MN rows of $\mathbf{A}_{0,0}$ in (3.27) can be denoted as $\mathrm{a}_{mN+n} = [\beta_1^{(km+n)} \quad \beta_2^{(km+n)} \cdots \beta_D^{(km+n)}]$ $m = 0, 1, \cdots M-1, n = 0, 1, \cdots, N-1$ Hence $\mathbf{A}_{0,0}$ consists of $k(M-1)+N$ distinct rows. Moreover these $k(M-1)+N$ rows make up a Vandermonde matrix of size $(N+(M-1)k) \times D$. Hence the rank of $\mathbf{A}_{0,0} = \min(D, k(M-1)+N)$. When $D > kM+N$, the source steering vectors become linearly dependent rendering $\mathbf{A}_{0,0}$ column rank deficient.

3.10.2 A Sufficient Condition on Unique Identifiability

Even when the array manifold is full column rank, it does not ensure unique identifiability in $2D$ DOA estimation. This is because the steering vector for a different $2D$ DOA pair (not belonging to any of the source DOAs), might lie in the subspace spanned by the manifold of the source DOAs and hence can produce a *false peak* in the null spectrum. The following examples illustrates this:

3.10.2.1 Example (Full column rank of $\mathbf{A}_{0,0}$ does not imply unique identifiability)

Assuming $D > N$, let us choose the D DOAs as per Lemma 3.10.1 by first including N distinct DOA pairs of the form

$$(\omega_{11}, \omega_{2i}) \quad i = 1, 2, \cdots N$$

and choosing the remaining $D-N$ DOA pairs at random from the remaining $MN-N$ distinct pairs given by lemma 3.10.1. Then, the array manifold is full column rank D and its orthogonal complement has rank $MN-D$. Now consider the pair $(\omega_{11}, \omega_{2,N+1})$. Clearly it does not correspond to any of the source DOA pairs. However, the $N+1$ DOA pairs $(\omega_{11}, \omega_{2i}), i = 1, 2, \cdots, N+1$, consist of $N+1$ distinct values of ω_2 paired with the same value of ω_1 (i.e., $\omega_1 = \omega_{1,1}$) (they lie on a vertical line $\omega_1 = \omega_{11}$ in the $(\omega_1 - \omega_2)$ plane). Hence, they correspond to the set of values given in (3.32) of Example 2 (here N_1 corresponds to $N+1$), and the steering vectors corresponding to these $N+1$ DOAs are linearly dependent. In other words, the steering vector corresponding to $(\omega_{11}, \omega_{1,N+1})$ lies in the range space of $\mathbf{A}_{0,0}$ and hence, it is orthogonal to the null space of the array manifold, yielding a false peak in the null spectrum.

Motivated by this counterexample, we now present a sufficient condition on the choice of D DOA pairs which will guarantee unique identifiability of the source DOAs. In the theorem,

$\mathbf{a}_{M,N}(\omega_1, \omega_2)$ denotes the steering vector of the $2D$ array (a column of $\mathbf{A}_{0,0}$) corresponding to a generic $2D$ DOA pair (ω_1, ω_2).

Theorem 3.10.1. *Let us consider $D \leq (M-1)(N-1)$ source DOA pairs which are chosen from the $(M-1)(N-1)$ DOA pairs (in accordance with Lemma 3.10.1)*

$$(\omega_{1i}, \omega_{2j}), i = 1, 2, \cdots, M-1, j = 1, 2, \cdots, N-1. \tag{3.33}$$

Then the MUSIC spectrum will exhibit a null if and only if $\mathbf{a}_{M,N}(\omega_1, \omega_2)$ corresponds to one of the D source steering vectors. □

Proof. Since the choice of these $D < MN$ source DOAs are in accordance with Lemma 3.10.1, the array manifold is of full column rank D. Let $\mathbf{E}_N \in \mathcal{C}^{MN \times (MN-D)}$ be the matrix whose columns span the $(MN-D)$ dimensional orthogonal complement of the range space of $\mathbf{A}_{0,0}$ [160]. Hence, $\mathbf{a}^H_{M,N}(\omega_{1i}, \omega_{2i})\mathbf{E}_N = \mathbf{0}^T$ for each $i = 1, 2, \cdots, D$ implying that the MUSIC spectrum will yield D peaks at the true location of the D DOAs. For the necessary condition, we need to ensure that $\mathbf{a}^H_{M,N}(\omega_1, \omega_2)\mathbf{E}_N \neq \mathbf{0}^T$ if (ω_1, ω_2) does not correspond to one of D source DOA pairs (i.e., the MUSIC spectrum does not produce a peak for any other value of (ω_1, ω_2) apart from the D DOA pairs). We will prove this by contradiction. If possible, let $\mathbf{a}^H_{M,N}(\omega_{1,M}, \omega_{2,N})\mathbf{E}_N = \mathbf{0}^T$ where $(\omega_{1,M}, \omega_{2,N})$ does not correspond to any of the source DOA pairs given in (3.33) (i.e., assume there is a false peak). Hence

$$\underbrace{\begin{bmatrix} \mathbf{a}^H_{M,N}(\omega_{11}, \omega_{21}) \\ \mathbf{a}^H_{M,N}(\omega_{12}, \omega_{22}) \\ \vdots \\ \mathbf{a}^H_{M,N}(\omega_{1,M}, \omega_{2,N}) \end{bmatrix}}_{\mathbf{A}^H_{D+1}} \mathbf{E}_N = \begin{bmatrix} \mathbf{0}^T \\ \mathbf{0}^T \\ \vdots \\ \mathbf{0}^T \end{bmatrix} \tag{3.34}$$

Here $\mathbf{A}_{D+1} \in \mathcal{C}^{MN \times (D+1)}$ corresponds to the manifold of the same array but with $D+1$ sources. Now, these $D+1 < MN$ sources are selected from the MN DOA pairs $\{(\omega_{1i}, \omega_{2j}), i = 1, 2, \cdots, M, j = 1, 2, \cdots, N\}$ and hence, by Lemma 3.10.1, $\mathbf{A}_{D+1}$ is of full column rank $D+1$. Hence the orthogonal complement of $\mathcal{R}(\mathbf{A}_{D+1})$ is of rank $MN-D-1$. However, according to (3.34), the range space of $\mathbf{E}_N$, which is of rank $MN-D$, belongs to the orthogonal complement of $\mathcal{R}(\mathbf{A}_{D+1})$. This is a contradiction since the orthogonal complement of $\mathcal{R}(\mathbf{A}_{D+1})$ is of rank $MN-D-1$. Hence (3.34)

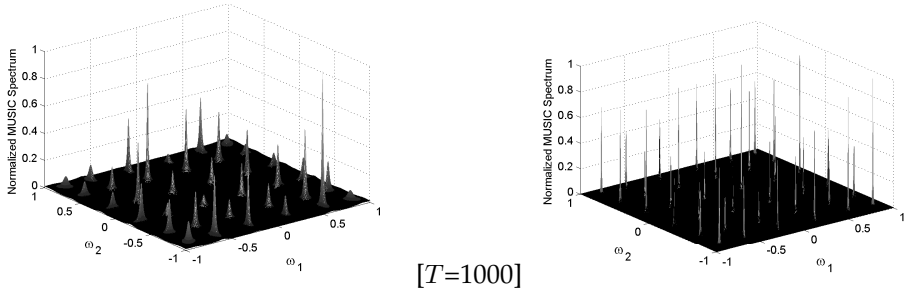
cannot hold true and hence $\mathbf{a}_{M,N}^H(\omega_1,\omega_2)\mathbf{E}_N \neq \mathbf{0}^T$ if (ω_1,ω_2) does not correspond to one of the impinging D DOA pairs. □

3.11 Simulations

In this section, we provide numerical examples to illustrate the superior performance of the nested array in terms of degrees of freedom for $2D$ DOA estimation and beamforming. We consider a randomly generated dense lattice generator $\mathbf{M}$, and $\mathbf{P}$ is constructed as an integer matrix whose $\mathbf{\Lambda}$ (obtained from Smith form of $\mathbf{P}$) is given by $\begin{pmatrix} 3 & 0 \\ 0 & 3 \end{pmatrix}$. We choose $N_1^{(\mathrm{s})} = 2$, $N_2^{(\mathrm{s})} = 3$, so there are in all $(2N_1^{(\mathrm{s})}+1)N_2^{(\mathrm{s})}+\det(\mathbf{\Lambda})-2 = 22$ sensors on the array (we exclude the sensor at $\mathbf{0}$ since $N_1^{(\mathrm{s})} > 1$ and $N_2^{(\mathrm{s})}, \lambda_1, \lambda_2 > 1$). Using the proposed algorithm on this nested array, we can construct $\mathbf{R}_{\mathrm{co-array}}$ corresponding to a larger 8×9 array with a total of 72 virtual sensors. We now consider $D = 36$ narrowband waves impinging on this array ($36 > 22 =$ number of physical sensors in the array) whose DOAs are chosen by generating 25 (ω_1,ω_2) pairs as $\Big\{\Big(\sin(-\pi/3+2(i-1)\pi/18), \sin(-\pi/3+2(k-1)\pi/18)\Big), i = 1,2,\cdots 6, k = 1,2,\cdots,6\Big\}$. It can be verified that these (ω_1,ω_2) pairs are chosen in accordance with the condition in Theorem 3.10.1 and hence they satisfy the condition for unique identifiability. Since the matrix $\mathbf{R}_{\mathrm{co-array}}$ corresponds to the virtual $2D$ array of size 8×9, we compare the performance of the proposed method with that of an actual larger $2D$ array of size 8×9 with sensors on the same lattice. We denote this as the "benchmark" case.

3.11.1 MUSIC spectra

Figure 3.18(a) and (b) shows the MUSIC spectrum obtained from $\mathbf{R}_{\mathrm{co-array}}$ for two different values of snaphsots $T = 100$, and 1000. Notice how the spectrum of the proposed method evolves as T increases, resolving all the 36 sources. For the given number of sources ($D = 36$), the subspace based methods could not have been applied on the physical array with 22 sensors (since $D > 22$). However, the MUSIC spectrum obtained from $\mathbf{R}_{\mathrm{co-array}}$ resolves all the 36 peaks like the benchmark array with more sensors, with the quality of the spectrum improving with increasing snapshots.



[T=100] [T=1000]

Figure 3.18: $2D$ MUSIC spectrum using the proposed algorithm for identifying $D = 36$ sources using the $2D$ nested array for two values of snapshots.

3.11.2 RMSE v/s SNR and snapshots

We now study the performance of the proposed method by studying the RMSE of the angle estimates as a function of both SNR and snapshots. We individually plot the RMSE in ϕ and θ as a function of SNR for $T = 500$ snapshots, for both the proposed method and the benchmark. The RMSE is computed by averaging over 800 Monte Carlo runs and averaged over all the 36 DOA pairs separately for ϕ and θ. The Monte Carlo run is performed by randomly generating signal and noise vectors for a given dense lattice generator $\mathbf{M}$. Fig. 3.19 shows the desired RMSE plot.

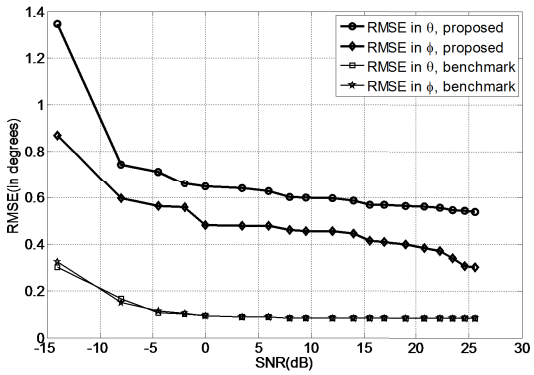


Figure 3.19: Comparison of RMSE (in degrees, averaged over all $D = 36$ DOA pairs) vs. SNR between the proposed DOA estimation algorithm applied on the $2D$ nested array, and traditional $2D$ MUSIC applied on the benchmark array. Here $T = 500$.

Next, in Fig. 3.20 we plot the RMSE of the proposed method and the benchmark for SNR=6 dB by varying the total number of snapshots T. The performance of the proposed method improves considerably with the number of snapshots, approaching the benchmark for large number of snapshots. It is to be noted that for finite snapshots, the proposed method only approximates

the benchmark model, and hence there is a gap between the performances of the two. This is especially true for the plot showing RMSE v/s SNR where the gap between the proposed method and the benchmark model remains non zero even as we increase SNR.

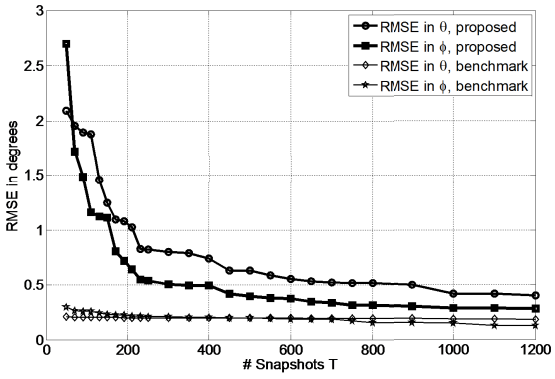


Figure 3.20: Comparison of RMSE (in degrees, averaged over all $D = 36$ DOA pairs) vs. the number of snapshots between the proposed DOA estimation algorithm applied on the $2D$ nested array, and traditional $2D$ MUSIC applied on the benchmark array. Here SNR$= 6$ dB.

3.11.3 Detection Performance

In the previous examples, it was assumed that the number of sources D is known. However, in practice, we need to estimate it from the available data. To do this, we would need to find some optimum detection algorithm suited to our model, which is a subject of our future research in this direction. However, we applied the threshold based technique in [41] which proved to give satisfactory results when applied to $\mathbf{R}_{\mathrm{co-array}}$. The estimated probability of detection of this proposed method at SNR$= 10$ dB as a function of snapshots is shown in Fig. 3.21. For comparison, we also plot the corresponding performance of the threshold technique applied to the benchmark $2D$ array. The probabilities were obtained by running 1000 Monte Carlo simulations averaged over another 1000 runs. We can see that the detection performance improves dramatically with increasing snapshots and approaches that of the benchmark asymptotically.

3.12 Conclusion

In this chapter, we have proposed a new class of $2D$ arrays which is a combination of $O(M + N)$ sensors distributed over two suitably related non separable lattices, whose difference co-array can

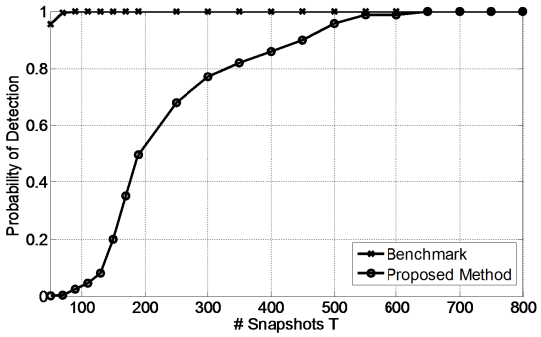


Figure 3.21: Comparison of probability of detecting all the $D = 36$ sources for the $2D$ nested array and the benchmark array, as a function of the number of snapshots. Here SNR$= 10$ dB.

give rise to a much larger $2D$ array with $O(MN)$ sensors on the dense lattice. This is the two dimensional extension and generalization of the $1D$ nested arrays proposed earlier. The dense lattice generator and the integer matrix $\mathbf{P}$ can be arbitrarily chosen. However, we proved that from the perspective of a specific requirement on the minimum distance in the lattice (due to mutual coupling etc.) the best geometry of the dense lattice used to tile the sparse lattice is given by a hexagonal geometry. Many of the issues related to the geometry of these arrays in $2D$ space have been discussed in great detail. It has also been shown that the Smith form decomposition of the integer matrix relating the two lattices provides a very useful viewpoint in the design of the $2D$ nested array. We further have demonstrated the practical applications of the $2D$ nested array in array processing problems such as $2D$ DOA estimation, when the number of sources can be possibly much larger than the number of physical sensors. The number of elements in the virtual array has been maximized (under the constraint of a fixed number of physical sensors) to yield closed form solutions for the sensor positions in the physical $2D$ nested array. One of the solutions to this problem is always shown to yield $1D$ manifolds in the physical array domain. In order to exploit the increased degrees of freedom offered by this virtual array, a novel algorithm based on $2D$ spatial smoothing has also been proposed and its performance successfully demonstrated through numerical examples.

3.13 Appendix

3.13.1 Proof of Theorem 3.7.2

Since the sensor at $\mathbf{0}$ is absent, all the virtual sensors in the cross difference co-array that are generated due to the zero sensor will be absent. In particular, this means virtual sensors located at $\pm(\tilde{\mathbf{N}}^{(\mathrm{s})}\tilde{\mathbf{n}}^{(\mathrm{s})} - \mathbf{0})$ and $\pm(\tilde{\mathbf{N}}^{(\mathrm{d})}\tilde{\mathbf{n}}^{(\mathrm{d})} - \mathbf{0})$ where $\tilde{\mathbf{n}}^{(\mathrm{s})}$ and $\tilde{\mathbf{n}}^{(\mathrm{d})}$ take up values given by (3.6) and (3.8) respectively, will be absent. Now we will argue that these can be regenerated by considering the individual self differences of the sparse array and the dense array. Note that self differences of the sparse array sensor locations are of the form

$$\tilde{\mathbf{N}}^{(\mathrm{s})}(\tilde{\mathbf{n}}_1^{(\mathrm{s})} - \tilde{\mathbf{n}}_2^{(\mathrm{s})}),$$

$$\tilde{\mathbf{n}}_1^{(\mathrm{s})}, \tilde{\mathbf{n}}_2^{(\mathrm{s})} \in [-N_1^{(\mathrm{s})} \quad N_1^{(\mathrm{s})}] \times [0 \quad N_2^{(\mathrm{s})} - 1], \tilde{\mathbf{n}}_1^{(\mathrm{s})}, \tilde{\mathbf{n}}_2^{(\mathrm{s})} \neq \mathbf{0}$$

Consider the subset of the sparse array given by

$$\tilde{\mathbf{N}}^{(\mathrm{s})}[k \quad \tilde{n}^{(\mathrm{s})}]^T$$

where

$$0 \leq \tilde{n}^{(\mathrm{s})} \leq N_2^{(\mathrm{s})} - 1, k \neq 0.$$

Since $N_1^{(\mathrm{s})} \geq 1$, the sparse array indeed contains this as a subset for $k \neq 0$. It can be easily verified that the self difference co-array of this set, for each $k \neq 0$, contains the differences $\pm\tilde{\mathbf{N}}^{(\mathrm{s})}[0 \quad \tilde{n}^{(\mathrm{s})}]^T$ where

$$0 \leq \tilde{n}^{(\mathrm{s})} \leq N_2^{(\mathrm{s})} - 1 \tag{3.35}$$

Similarly considering the subset $\{\tilde{\mathbf{N}}^{(\mathrm{s})}[\tilde{n}^{(\mathrm{s})} \quad k]^T, -N_1^{(\mathrm{s})} \leq \tilde{n}^{(\mathrm{s})} \leq N_1^{(\mathrm{s})}, k \neq 0\}$ (since $N_2^{(\mathrm{s})} > 1$, this is indeed a subset of the sparse array for $k \neq 0$), it can be verified that the self difference co-array

of this subset contains, for each $k \neq 0$, the difference set

$$\pm\tilde{\mathbf{N}}^{(\mathrm{s})}[\tilde{n}^{(\mathrm{s})} \quad 0]^T, -N_1^{(\mathrm{s})} \leq \tilde{n}^{(\mathrm{s})} \leq N_1^{(\mathrm{s})}. \tag{3.36}$$

Now, consider two subsets of the sparse array:

$$\tilde{\mathbf{N}}^{(\mathrm{s})}[k \quad \tilde{n}_{21}^{(\mathrm{s})}]^T, 0 \leq \tilde{n}_{21}^{(\mathrm{s})} \leq N_2^{(\mathrm{s})} - 1, k \neq 0$$

and

$$\tilde{\mathbf{N}}^{(\mathrm{s})}[0 \quad \tilde{n}_{22}^{(\mathrm{s})}]^T, 1 \leq \tilde{n}_{22}^{(\mathrm{s})} \leq N_2^{(\mathrm{s})} - 1$$

The cross differences of these two sets produce sensors located at

$$\pm\tilde{\mathbf{N}}^{(\mathrm{s})}[k \quad \tilde{n}_2^{(\mathrm{s})}]^T, -N_2^{(\mathrm{s})} + 1 \leq \tilde{n}_2^{(\mathrm{s})} \leq N_2^{(\mathrm{s})} - 2$$

Repeating this for each k in the range $-N_1^{(\mathrm{s})} \leq k \leq N_1^{(\mathrm{s})}, k \neq 0$, it can be seen that all sensors of the form $\tilde{\mathbf{N}}^{(\mathrm{s})}[\tilde{n}_1^{(\mathrm{s})} \quad \tilde{n}_2^{(\mathrm{s})}]^T$ where

$$N_1^{(\mathrm{s})} \leq |\tilde{n}_1^{(\mathrm{s})}| \leq 1, \quad -N_2^{(\mathrm{s})} + 1 \leq \tilde{n}_2^{(\mathrm{s})} \leq N_2^{(\mathrm{s})} - 1$$

can be generated. This, together with the differences given by (3.35) and (3.36) produces all virtual sensors located at $\tilde{\mathbf{N}}^{(\mathrm{s})}\tilde{\mathbf{n}}^{(\mathrm{s})}$ where $\tilde{\mathbf{n}}^{(\mathrm{s})}$ takes up all the values given by (3.6), thereby generating all sensor locations on the original sparse physical array (and their negated values).

By a similar argument on the self differences of the dense lattice sensors (by replacing $N_1^{(\mathrm{s})}$ with $(\lambda_1 - 1)/2$ and $N_2^{(\mathrm{s})}$ with λ_2 and using the assumption $\lambda_1, \lambda_2 > 1$), it can be shown that all virtual locations given by $\tilde{\mathbf{N}}^{(\mathrm{d})}[\tilde{n}_1^{(\mathrm{d})} \quad \tilde{n}_2^{(\mathrm{d})}]^T$ where

$$-(\lambda_1 - 1)/2 \leq \tilde{n}_1^{(\mathrm{d})} \leq (\lambda_1 - 1)/2$$

$$-(\lambda_2 - 1) \leq \tilde{n}_2^{(\mathrm{d})} \leq \lambda_2 - 1$$

can be generated. Hence, we can exclude the sensor at $\mathbf{0}$ and yet gain back all the virtual sensors

thus lost in the cross difference co-array by considering the individual self differences of the two arrays.

3.13.2 Proof of Theorem 3.8.1 when K is odd

We prove 3.8.1 for the case when K is odd. The proof for even K can also be obtained on similar lines.

Proof. Without loss of generality we can assume that

$$N_0 N_2^{(\mathrm{s})} = n - \delta \quad \lambda_1 \lambda_2 = n + \delta \tag{3.37}$$

where

$$n = (K+1)/2$$

and δ is an integer with $0 \leq |\delta| < n$. Also, we have

$$N_0 \lambda_1 = N_2^{(\mathrm{s})} \lambda_2 - \Delta \tag{3.38}$$

where Δ is an integer. Notice

$$N_0 \lambda_1 N_2^{(\mathrm{s})} \lambda_2 = (n+\delta)(n-\delta)$$

Using (3.38) and denoting $N_0 \lambda_1 = x$ (note that x needs to be odd to satisfy the constraints) we get

$$x(x + \Delta) = n^2 - \delta^2 \tag{3.39}$$

Solving this quadratic equation for x yields

$$x = \frac{-\Delta + \sqrt{\Delta^2 + 4n^2 - 4\delta^2}}{2}$$

The integers Δ and δ need to be such that this gives an odd integer solution for x. Now the objective function to be optimized can be rewritten as $\text{DOF}(\Delta, \delta) = 2x(x+\Delta) - x$
$= 2(n^2 - \delta^2) + \frac{\Delta - \sqrt{\Delta^2 + 4n^2 - 4\delta^2}}{2}$. which is a function of Δ and δ only. For a given Δ, it can be easily

verified that (3.13.2) is a *decreasing function of* δ^2. So for a given Δ, to maximize DOF, we need to find the smallest δ^2 which gives rise to an odd integer solution for x in (3.39).

3.13.2.1 $K+1 = 4p+2$

In this case $n = 2p+1$ is odd. Following strategy S_Δ, we first consider $\Delta = 0$. Now for this Δ, $\delta = 0$ maximizes the DOF but we need to verify that it indeed gives a feasible solution for odd integer x in (3.39). For this, we need to ensure that this choice of Δ and δ yields an odd integer solution for x in (3.39). Since n is odd, $\Delta = 0$, $\delta = 0$ indeed yields a feasible x and hence this corresponds to the optimal solution. So we have

$$x = N_0\lambda_1 = n = 2p+1, N_2^{(\mathrm{s})}\lambda_2 = x + \Delta = 2p+1$$

and

$$N_0 N_2^{(\mathrm{s})} = n - \delta = 2p+1, \lambda_1\lambda_2 = n + \delta = 2p+1.$$

The optimal value of the DOF is given by $(K+1)^2/2 - (K+1)/2$.

3.13.2.2 $K+1 = 4p$

Here $n = 2p$ is even, and both Δ and δ cannot be made 0 in (3.39) because x needs to be an odd integer. Let us investigate the following cases:

1. $\Delta = 0, \delta \neq 0$:

   In this case, $x^2 + \delta^2 = n^2$. Hence x, δ and n should form a Pythagoren triplet. For given n, there might not exist solutions for odd x and an integer δ satisfying this. However, the degrees of freedom $\mathrm{DOF}(\Delta = 0, \delta)$ in this case is bounded by $\mathrm{DOF}(\Delta = 0, \delta \neq 0)$
   $< DOF(\Delta = 0, \delta \pm 1)$
   $= (K+1)^2/2 - 2 - \sqrt{(K+1)^2/4 - 1}$
   $< (K+1)^2/2 - (K+1)/2 - 1$ Here the first inequality is due to the fact that for a given Δ, $DOF(\Delta, \delta)$ is a decreasing function of δ^2 and the second inequality can be readily verified by noting that the second quantity is greater than the first.

2. $|\Delta| = 1$: In this case, $x^2 \pm x + \delta^2 = n^2$. Here also, for a given n, the equation need not be

solvable for all values of δ. For example, $\delta = 0$ gives no solution for x since $x(x \pm 1) = n^2$ has no integer solution (since a perfect square cannot be a product of consecutive integers). However, whenever a solution exists, it is bounded by DOF($\Delta = \pm 1, \delta \neq 0$)

$< DOF(\Delta = \pm 1, \delta = \pm 1)$

$= \frac{(K+1)^2}{2} - 2 - \frac{\pm 1 + \sqrt{(K+1)^2 - 3}}{2}$

$< \frac{(K+1)^2}{2} - 2 - \frac{\pm 1 + ((K+1)-1)}{2}$

$\leq \frac{(K+1)^2}{2} - \frac{(K+1)}{2} - 1$

3. $|\Delta| = 2$

   Here, $x^2 \pm 2x + \delta^2 = n^2$. Again, for $\delta = 0$, no solution exists since solving for x does not yield an integer solution for any n. However, putting $\delta = \pm 1$ gives $x^2 \pm 2x + 1 = n^2$, i.e., $(x \pm 1)^2 = n^2$. Since x is odd and n is even, this always gives feasible solution! Hence, the minimum Δ and δ for which a solution exists for all n, is

$$|\Delta| = 2, \quad \delta = 1.$$

   (Note that the DOF is a function of only δ^2). The DOF for $\Delta = 2$ is given by DOF($\Delta = 2, \delta = 1$)

   $= (K+1)^2/2 - 2 - (K+1)/2 + 1$

   $= (K+1)^2/2 - (K+1)/2 - 1$ which is strictly greater than DOF($\Delta = 0, \delta$) and DOF($|\Delta| = 1, \delta$). The DOF for $\Delta = -2$ can be shown to be strictly less than that for $\Delta = 2$. Hence $\Delta = 2$, $\delta = 1$ are the smallest possible values of Δ and δ which always lead to a feasible solution and also the DOF thus obtained is strictly greater than the DOF obtainable for any other smaller values of $|\Delta|$. Hence for $n = 2p$ (i.e., $K + 1 = 4p$) it provides the optimum solution. The value of $x = N_0\lambda_1$ is given by

$$N_0\lambda_1 = n - 1 = 2p - 1$$

   and using (3.38) and (3.37), we get

$$\lambda_2 N_2^{(s)} = x + 2 = n + 1 = 2p + 1$$

and

$$\lambda_1\lambda_2 = n + \delta = 2p + 1, N_0 N_2^{(\mathrm{s})} = n - \delta = 2p - 1$$

□

3.13.3 Proof of Theorem 3.8.2

Let $N_1^{(\mathrm{s})} = \tilde{N}_1, N_2^{(\mathrm{s})} = \tilde{N}_2, \lambda_1 = \tilde{\lambda}_1, \lambda_2 = \tilde{\lambda}_2$ denote a feasible solution. Then we can form another feasible solution $N_2^{(\mathrm{s})} = \tilde{N}_2(2\tilde{N}_1 + 1)$, $N_1^{(\mathrm{s})} = 0, \lambda_2 = \tilde{\lambda}_1\tilde{\lambda}_2$, $\lambda_1 = 1$ (it can be verified that these values satisfy the constraints and hence indeed form a feasible point). However substituting these values in the objective (3.13) strictly increases the value of the objective function since

$$2(2\tilde{N}_1 + 1)\tilde{N}_2\tilde{\lambda}_1\tilde{\lambda}_2 - (2\tilde{N}_1 + 1)\tilde{\lambda}_1 < 2(2\tilde{N}_1 + 1)\tilde{N}_2\tilde{\lambda}_1\tilde{\lambda}_2 - 1$$

Hence, in the optimal solution, we will always have $2N_1^{(\mathrm{s})} + 1 = \lambda_1 = 1$. Note that this value of λ_1 automatically satisfies the second and third constraints of (3.13). Therefore, we can recast the problem (3.13) as

$$\max_{N_2^{(\mathrm{s})}, \lambda_2 \in \mathbb{N}^+} 2N_2^{(\mathrm{s})}\lambda_2 - 1 \tag{3.40}$$

subject to:

$$N_2^{(\mathrm{s})} + \lambda_2 = K + 1.$$

which can be readily solved using $AM - GM$ inequality. This yields the solution

$$N_2^{(\mathrm{s})} = \lambda_2 = (K+1)/2, \quad K \text{ is odd}$$
$$N_2^{(\mathrm{s})} = (K+2)/2, \quad \lambda_2 = K/2, \quad K \text{ is even.}$$

The maximized degrees of freedom in the virtual array turns out to be

$$DOF = \begin{cases} (K+1)^2/2 - 1, & K \text{ is even} \\ (K+1)^2/2 - 3/2, & K \text{ is odd.} \end{cases}$$

□

Chapter 4

Higher Order Statistics and Extension of Nested Arrays

4.1 Introduction

Direction of arrival (DOA) estimation algorithms, based on the fourth order cumulants of non Gaussian source signals have been developed for array processing applications for around two decades [162, 50, 63, 154], particularly for the advantages they offer over traditional DOA estimation methods which exploit the second order statistics of the data. These advantages include (1) ability to process more than $N-1$ statistically independent sources using N physical sensors, (2) higher resolution, (3) better robustness to modeling errors, and (4) better robustness to presence of strong background colored noise with unknown covariance [154, 44], and so forth.

Recently, these algorithms were extended to the case of processing an arbitrary even order ($2q$) cumulant of the data [44, 14], giving rise to a new family of DOA estimation algorithms, namely the $2q$ MUSIC. It has been shown that the identifiability, resolution performance, and robustness of these methods increase with q, thereby further improving the processing power of the fourth order cumulant based methods (which correspond to $q = 2$)[44]. The higher processing power of these methods is due to a deeper connection of the $2q$th order cumulants with the concept of a higher order virtual array [42] which behaves like an array with many more elements compared to the physical array. The connection of a longer virtual array with higher order statistics was first pointed out in [63],[43],[162] in the context of fourth order cumulants. This increased number of virtual sensors in this array is responsible for the higher processing power of the $2q$th order cumulant based methods. In [42], it has been pointed out that for $q \geq 2$, the number of virtual

sensors depends on the specific arrangement of the $2q$th order cumulant matrix. The best possible arrangement leads to a virtual array with $O(N^q)$ sensors and hence the $2q$th order cumulant based methods can identify $O(N^q)$ statistically independent sources [44].

In this chapter, we re-examine the virtual array associated with the $2q$th order cumulants and show that an even longer virtual array can be extracted from the cumulant matrices. This virtual array is shown precisely to be the $2q$th order difference co-array (defined in Section 4.4) of the physical array. For $q = 1$, second order difference co-arrays have been studied in the past [89, 149, 148, 126, 1, 2, 135, 184] and several constructive procedures for non uniform physical arrays have been proposed which give rise to second order difference co-arrays which contain ULA segments with $O(N^2)$ elements [159, 148, 155, 110, 135, 184]. Unlike the virtual arrays exploited in [44],[42], the number of elements in the $2q$th order difference co-array are independent of any particular arrangement of the cumulant matrix and depends only on the geometry of the physical array. Also, it can have $O(N^{2q})$ virtual sensors as opposed to the maximum of $O(N^q)$ virtual sensors obtainable in [44], and thereby can potentially enable one to identify $O(N^{2q})$ sources using the $2q$th order cumulants. However, for $q \geq 2$, linear arrays whose $2q$th order difference co-array contains $O(N^{2q})$ virtual sensors in the form of a ULA, have not been reported in the past. In order to bridge this gap, we propose a new family of non uniform arrays, parameterized by q, called the $2q$ level nested array, whose $2q$th order difference co-array contains a ULA with $O(N^{2q})$ elements. For $q = 1$, we proposed the 2 level nested array in [135] and in Chapter 3 and the concept of higher level nested array was also briefly introduced. However, we did not consider the $2q$th order difference co-array of the multiple level nested array because we were interested in second order statistics. The second order difference co-array of the multiple level nested array did not contain an ULA with $O(N^2)$ elements and hence it was not used for second order statistic based DOA estimation. However, we will show that the $2q$th order difference co-array of a $2q$ level nested array indeed contains an ULA with $O(N^{2q})$ elements, rendering it a very appropriate geomtery for algortihms based on $2q$th order cumulants. Also, in order to exploit the increased degrees of freedom offered by the $2q$ level nested array, we generalize a previously proposed DOA estimation algorithm [135] to process $2q$th order cumulants (for arbitrary q) and identify upto $O(N^{2q})$ sources. We prove that the proposed algorithm achieves the maximum identifiability among all DOA estimation methods using $2q$th order cumulants.

The content of this chapter is mainly drawn from [140] and part of it has been presented in [139].

4.2 Outline

The chapter is organized as follows. In Section 5.3, we briefly review the role of $2q$th order cumulants in DOA estimation, defining important terminology. In Section 4.4, we revisit the virtual array concept inherent in $2q$th order cumulant based array processing applications and show that an even longer virtual array, which is related to the $2q$th order difference co-array, can be extracted from the cumulant based model. In Section 4.5, we propose a new class of $2q$ level nested arrays whose $2q$th order difference co-array has $O(N^{2q})$ elements in the form of a ULA. In Section 4.6, we solve a combinatorial optimization problem to find the optimal distribution of sensors among the $2q$ levels which maximizes the degrees of freedom in the $2q$th order difference co-array. In Section 4.7, we propose a new DOA estimation algorithm which can exploit the $O(N^{2q})$ degrees of freedom offered by the $2q$ level nested array and identifies $O(N^{2q})$ sources. In Section 4.8, we prove that the proposed method can achieve the maximum identifiability that can be obtained by using $2q$th order cumulants. We also study the Cramer Rao bounds for the non Gaussian signal model when the number of sources can be less than the number of sensors. Finally in Section 4.9, we present several numerical simulation results to validate the proposed method and compare the proposed algorithm with existing $2q$ MUSIC based algorithms by studying the RMSE performance with respect to snapshots, SNR, modeling errors and probability of resolution. Section 4.10 concludes the chapter.

4.3 Review of $2q$th order cumulants and their role in array processing

Fourth order cumulant based DOA estimation methods have been elaborately treated for about two decades in [63],[154],[162],[50] and the references therein. An excellent tutorial on higher order cumulants is given in [124]. Recently, the application of $2q$th order cumulants for arbitrary integer $q \geq 2$ in the problem of DOA estimation was introduced in [44],[42],[14]. We review the main existing results in this section to build up the necessary background. The material in this section is taken mostly from [124] and [44].

4.3.1 Data Model

We consider a linear array of N sensors and D narrowband sources impinging on them with elevations angles $\{\theta_1, \theta_2, \cdots, \theta_D\}$. The sensors are assumed to be identical, omnidirectional, unpolarized and isotropic. The sources are assumed to be zero mean stationary non Gaussian sources which are *statistically independent*. While the formulation in [44] does not require the sources to be all statistically independent, this assumption is essential for our proposed method as will be seen later. The received signal vector at the output of the N sensors is given by

$$\mathbf{y}(t) = \sum_{d=1}^{D} \mathbf{a}(\theta_d) s_d(t) + \mathbf{u}(t) \tag{4.1}$$

where $\mathbf{u}(t)$ denotes the $N \times 1$ zero mean stationary Gaussian noise vector, statistically independent from the source signals $s_d(t), d = 1, 2, \cdots, D$. The vector $\mathbf{a}(\theta_d)$ denotes the $N \times 1$ steering vector for the dth source direction, and its nth component is given by

$$a_n(\theta_d) = e^{j\frac{2\pi}{\lambda} x_n \sin\theta_d} \tag{4.2}$$

where x_n denotes the distance of the nth sensor in the array from the origin.

4.3.2 $2q$th order circular cumulants

DOA estimation algorithms in [44] are based on computing the $2q$th order circular cumulants of the received signal vector $\mathbf{y}(t)$ [44]. More specifically, denoting $\mathbf{y}(t) = [y_1(t) \quad y_2(t) \cdots y_N(t)]^T$, the $2q$th order circular cumulants of the random variables $y_{i_1}(t), y_{i_2}(t), \cdots, y_{i_{2q}}(t)$ denoted as $\mathrm{Cum}[y_{i_1}(t), y_{i_2}(t), \cdots, y_{i_q}(t), y^*_{i_{q+1}}(t), y^*_{i_{q+2}}(t), \cdots, y^*_{i_{2q}}(t)]$ are computed, for $1 \le i_l \le N, 1 \le l \le 2q$. The circular cumulants are explicitly given by the Leonov-Shiryaev formula [108] as follows:

$$\mathrm{Cum}[y_{i_1}^{\varepsilon_1}(t), y_{i_2}^{\varepsilon_2}(t), \cdots, y_{i_{2q}}^{\varepsilon_{2q}}(t)] \tag{4.3}$$

$$= \sum_{k=1}^{2q} (-1)^{k-1}(k-1)! \sum_{m} \mathrm{E}\Big[\prod_{l \in S1^{(m)}} y_{i_l}^{\varepsilon_l}(t)\Big] \times \mathrm{E}\Big[\prod_{l \in S2^{(m)}} y_{i_l}^{\varepsilon_l}(t)\Big] \tag{4.4}$$

$$\times \cdots \times \mathrm{E}\Big[\prod_{l \in Sk^{(m)}} y_{i_l}^{\varepsilon_l}(t)\Big] \tag{4.5}$$

Here $(S1^{(m)}, S2^{(m)}, \cdots, Sk^{(m)})$ denotes a particular (the mth) partition in k sets of the set of integers $(1, 2, \cdots, 2q)$, and

$$y_{i_l}^{\varepsilon_l}(t) = y_{i_l}(t), l = 1, 2, \cdots, q$$

and

$$y_{i_l}^{\varepsilon_l}(t) = y_{i_l}^*(t), l = q+1, \cdots, 2q.$$

In practice, assuming ergodicity, the cumulants are estimated from L samples of the received vector $\mathbf{y}(l) \triangleq \mathbf{y}(lT_s), 0 \leq l \leq L-1$ where T_s is the sampling period, by replacing the expected values in (4.5) by their empirical values averaged over L samples. This estimate is asymptotically unbiased and consistent [44].

The DOA estimation methods based on higher order statistics which were presented earlier in [44, 14], arrange the cumulants in the form of a $N^q \times N^q$ $2q$th order Hermitian matrix which is proved to possess the required structure for application of subspace based DOA estimation algorithms like MUSIC (rather, an extension of it for $2q$ order statistics of the data, namely the $2q$ MUSIC). An important result proved in [44] shows that for $q > 1$, the cumulants can be arranged in more than one way to give rise to cumulant matrices with different algebraic structures, which lead to different identifiability and resolution properties of $2q$ MUSIC algorithm. In particular, given q, the $2q$ th order cumulants can be arranged in the form of cumulant matrice(s) in q different ways. These matrices are indexed by l, also called the orientation, $(0 \leq l \leq q-1)$ and denoted as $\mathbf{C}_{2q,\mathbf{y}}(l)$. Under the assumption of statistically independent source signals and white Gaussian noise in the data model, the matrix $\mathbf{C}_{2q,\mathbf{y}}(l)$ is given by $\mathrm{C}_{2q,\mathbf{y}}(l) = \sum_{d=1}^{D} c_{2q,s_d}[\mathbf{a}(\theta_d)^{\otimes l} \otimes \mathbf{a}(\theta_d)^{*\otimes(q-l)}]$
$\times [\mathbf{a}(\theta_d)^{\otimes l} \otimes \mathbf{a}(\theta_d)^{*\otimes(q-l)}]^H + \sigma_u^2 \mathbf{I}_{N^q \times N^q} \delta(q-1)$ where

$$c_{2q,s_d} = \mathrm{Cum}[s_{d_1}(t), \cdots, s_{d_q}(t), s_{d_{q+1}}^*(t), \cdots, s_{d_{2q}}^*(t)] \tag{4.6}$$

with $d_j = d, 1 \leq j \leq 2q$. The noise power is denoted by σ_u^2 and $\mathbf{a}(\theta_d)^{\otimes l}$ is the $N^l \times 1$ vector given by

$$\mathbf{a}^{\otimes l} \triangleq \underbrace{\mathbf{a}(\theta_d) \otimes \mathbf{a}(\theta_d) \otimes \cdots \otimes \mathbf{a}(\theta_d)}_{l \quad \text{times}}.$$

The noise is assumed white Gaussian, so the $2q$th order cumulants of the noise for $q \geq 2$ are zero, giving rise to the $\sigma_u^2 \mathbf{I}_{N^q \times N^q} \delta(q-1)$ term in (4.3.2). This expression for $\mathbf{C}_{2q,\mathbf{y}}(l)$ can be derived by first computing the N^{2q} possible cumulant values using (4.5) by substituting the values of y_k ($1 \leq k \leq N$) from the data model given by (4.1) and then using the properties of cumulants given in [124] regarding sums of statistically independent random variables, Gaussian random variables, and scaled random variables. Then these cumulant values can be arranged in the form of a $N^q \times N^q$ Hermitian matrix in q different ways (details in [44]), the lth ($0 \leq l \leq q-1$) of which is given by (4.3.2). The reason why the cumulant matrices for different l can give rise to different processing power (identifiability) and resolution when subspace based DOA estimation algorithms are applied on them, is because of their connection with the concept of a *virtual* array associated with the original array, as described in the next section.

4.3.3 Virtual Array Associated with $2q$th Order cumulants

The $2q$ th order cumulant based DOA estimation methods in [42] show the connection between the structure of the cumulant matrix and the concept of virtual array which is the reason behind the capability of the cumulant based methods to identify many more sources than sensors. Such a connection was first pointed out for $q = 2$ (fourth order cumulants) in [63, 154, 162]. Consider the vector

$$\mathbf{a}_{virtual}^{(l)}(\theta) \triangleq \mathbf{a}(\theta)^{\otimes l} \otimes \mathbf{a}(\theta)^{*\otimes(q-l)} \tag{4.7}$$

appearing in the cumulant matrix $\mathbf{C}_{2q,\mathbf{y}}(l)$. The elements of this vector are given by

$$e^{j\frac{2\pi}{\lambda}(\sum_{i=1}^{l} x_{n_i} - \sum_{i=l+1}^{q} x_{n_i})\sin\theta} \tag{4.8}$$

where each x_{n_i} takes values from the set of physical array locations $\{x_n, n = 1, 2, \cdots, N\}$ for $1 \leq i \leq q$, yielding a total of N^q elements (not all distinct) in the vector $\mathbf{a}_{virtual}^{(l)}(\theta)$. Comparing with (4.2), we find that (4.8) represents the steering vector of a Higher-Order (HO) virtual array [42] of N^q virtual sensors whose locations, denoted as $x_{n_1,n_2,\cdots,n_q}^{(l)}$ for $1 \leq n_i \leq N, \quad 1 \leq i \leq q$ are given by

$$x_{n_1,n_2,\cdots,n_q}^{(l)} = \sum_{i=1}^{l} x_{n_i} - \sum_{i=l+1}^{q} x_{n_i}. \tag{4.9}$$

The following remarks apply to the HO virtual array given by (4.9):

- The virtual array positions take up N^q values, however, not all of them are distinct. The number of distinct virtual sensors, denoted henceforth as $N_{2q}^{(l)}$ depends on the geometry of the physical array, i.e., the physical locations $\{x_n, n = 1, \cdots, N\}$. For given N, q, l, the quantity $N_{2q}^{(l)}$ is upper bounded by a quantity $N_{\max}[2q, l]$ which is tabulated for different values of q and l in Table 1 in [44, 42].

- The orientation l of the cumulant matrix plays an important role in the number of virtual sensors. For a given physical array, the virtual array for each l can be potentially different. For a given q and N, the upper bound $N_{\max}[2q, l]$ depends on l and it is maximum for $l = q/2$ if q is even and $l = (q + 1)/2$ for q odd [42].

- The upper bound $N_{\max}[2q, l]$, for a given q and l, corresponds to, in most cases, arrays with no particular symmetry. It cannot be reached if certain symmetry conditions are imposed on the physical array. For example, for a ULA with N sensors, $N_{2q}^{(l)} = q(N-1)+1 < N_{\max}[2q, l]$.

4.4 A New Look into the Virtual Array Associated with $2q$th order Cumulants

The connection of the $2q$th order cumulant matrices with virtual arrays leads to identification of upto $N_{2q}^{(l)} - 1$ independent sources, when the $2q$ MUSIC algorithm [44] is applied on the cumulant matrix $\mathbf{C}_{2q,\mathbf{y}}(l)$. However, in this chapter, we will show that this can be improved even further by exploiting the cumulant values in a different fashion, inspired from our earlier work in [135] for second order statistics (corresponding to $q = 1$). In order to establish our results we will first need to define a related concept of higher order difference co-array using which we will unify all possible arrangements of the cumulant matrix (corresponding to different values of l) to a fundamental higher order difference co-array, namely the $2q$th order difference co-array. As will be demonstrated, the DOA estimation method proposed in this chapter can exploit the increased degrees of freedom offered by this higher order co-array whereas $2q$ MUSIC is able to exploit a lower order (namely the qth order difference co-array assumin q is even) virtual array.

4.4.1 Higher Order Difference Co-Array

Definition 4.1: Consider a set of N sensors with position vectors $\mathbf{v}_j, j = 1, \cdots, N$. The $2q$th ($q \geq 1$ is an integer) order difference co-array of this set of sensors is defined as the set of distinct vectors from the set

$$\{\sum_{i=1}^{q} \mathbf{v}_{n_i} - \sum_{i=q+1}^{2q} \mathbf{v}_{n_i}, 1 \leq n_i \leq N\} \tag{4.10}$$

□

The difference set for $q = 1$, also known as the difference co-array, has been considered in the past in relation to MRA [1, 126],[149], 2 level nested array [135], coprime arrays [184] and direction finding methods related to fourth order cumulants [63, 154]. However, the connection between the $2q$th order cumulants and the $2q$th order difference set for any integer $q \geq 2$ has not been pointed out earlier.

In the $2q$th order difference set, each $\mathbf{v}_{n_i}$ takes up all possible values from the set $\{\mathbf{v}_j, j = 1, 2, \cdots, N\}$ and a total of N^{2q} values are generated. However, not all of them are distinct, and hence the difference co-array generally has $< N^{2q}$ elements, but nevertheless it can be seen from [42] that the maximum possible number of values is $O(N^{2q})$. The following points are worth noticing about the $2q$th order difference co-array:

- The $2q$th order difference co-array is symmetric about $\mathbf{0}$, i.e., for every element $\mathbf{v}$, there exists a corresponding element at $-\mathbf{v}$. Hence the positive half and negative half of the $2q$th order difference co-array has same number of elements.

- The $2q$th order difference co-array of an ULA with N sensors is another ULA with $2q(N-1)+1$ sensors. But it grows linearly with N as opposed to exponentially (e.g., $O(N^{2q})$).

- *Relation to MRA*: The $2q$th order difference co-array of an arbitrary linear array need not be a ULA in general. Moreover, it might not even have a subarray which is a ULA with $O(N^{2q})$ elements. Such arrays are difficult to construct and have not been studied for $q \geq 2$. For $q = 1$, there have been several systematic procedures to construct physical arrays whose second order difference set is a ULA with $O(N^2)$ elements [159, 148, 155, 110, 184] including the 2 level nested array proposed in [135]. Of particular interest is the minimum redundancy array [126] which is the physical array whose difference co-array is a ULA with maximum

possible elements. However, there is no known method to construct such arrays for every N and even the degrees of freedom is not known. For $q \geq 2$, the literature lacks in any theory about constructing arrays whose $2q$th order difference co-array is a ULA or contains an ULA with $O(N^{2q})$ sensors. So in this chapter, we provide a first hand attempt at developing a systematic and simple method to construct such arrays.

4.4.2 2qth order difference co-array and cumulants

Referring to (4.9), we find that for $l = q/2$ (assuming q is even), the virtual array corresponds to a qth order difference co-array. For a given even q, this is also the value of l for which the number of elements in the virtual array is maximized. In that case, the number of virtual sensors is $O(N^q)$.

For each l, now consider the vectorized cumulant matrix

$$\mathbf{c}_{vec}(l) = vec[\mathbf{C}_{2q,\mathbf{y}}(l)]. \tag{4.11}$$

In the following theorem, we will show that this vectorization [118, 135] of the cumulant values (instead of arranging them in the form of $N^q \times N^q$ matrix), leads to a model which captures the manifold of a much longer array, namely, the $2q$th order difference set of the original array, which is shown to be fundamentally related to the $2q$th order cumulant values of the received signal. This happens irrespective of the orientation (l) of $\mathbf{C}_{2q,\mathbf{y}}(l)$ (and the virtual array associated with that orientation). It also provides the first step towards building an appropriate $O(N^{2q}) \times O(N^{2q})$ matrix on which subspace based DOA estimation methods can be applied. We consider $q \geq 2$, so the noise term $\sigma_u^2 \mathbf{I}_{N^q \times N^q} \delta(q-1)$ in $\mathbf{C}_{2q,\mathbf{y}}(l)$ disappears.

Theorem 4.4.1. *The vectorized model in (4.11) can be expressed as*

$$\mathbf{c}_{vec}(l) = \mathbf{B}_{2q}(\boldsymbol{\theta})\mathbf{p} \tag{4.12}$$

where $\mathbf{B}_{2q}(\boldsymbol{\theta}) \in \mathcal{C}^{N^{2q} \times D}$ *(*$\boldsymbol{\theta} = [\theta_1 \quad \theta_2 \cdots \theta_D]^T$*) represents,* for all l*, the manifold of a virtual array corresponding to the* $2q$*th order difference co-array of the physical array and* $\mathbf{p} \in \mathcal{C}^{D \times 1}$ *represents the vector of the* $2q$*th order cumulants (*c_{2q,s_d}*,* $1 \leq d \leq D$*) of the* D *sources.* ♢ □

Proof. From (4.3.2) and (4.7), we get

$$\mathbf{c}_{vec}(l) = vec(\sum_{d=1}^{D} c_{2q,s_d} \mathbf{a}_{virtual}^{(l)}(\theta_d)\mathbf{a}_{virtual}^{(l)}(\theta_d)^H) \tag{4.13}$$

$$= \sum_{i=1}^{D} c_{2q,s_d} \mathbf{a}_{virtual}^{2q}(\theta_d) \tag{4.14}$$

where

$$\mathbf{a}_{virtual}^{2q}(\theta_d) = \mathbf{a}_{virtual}^{(l)*}(\theta_d) \otimes \mathbf{a}_{virtual}^{(l)}(\theta_d)$$

Hence the elements of $\mathbf{a}_{virtual}^{2q}(\theta_d)$ are given by ($1 \leq i \leq 2q, \quad 1 \leq n_i \leq N$)

$$e^{j\frac{2\pi}{\lambda}(\sum_{i=1}^{l} x_{n_i} - \sum_{k=l+1}^{q} x_{n_k} - \sum_{j=q+1}^{q+l} x_{n_j} + \sum_{m=q+l+1}^{2q} x_{n_m})\sin\theta_d} \tag{4.15}$$

$$= e^{j\frac{2\pi}{\lambda}(\sum_{i=1}^{q} w_i - \sum_{k=q+1}^{2q} w_k)\sin\theta_d} \tag{4.16}$$

where

$$w_i = \begin{cases} x_{n_i}, & 1 \leq i \leq l, q+1 \leq i \leq q+l \\ x_{n_{i+q}} & l+1 \leq i \leq q \\ x_{n_{i-q}} & q+l+1 \leq i \leq 2q \end{cases}$$

Since for each $1 \leq i \leq 2q$, x_{n_i} takes up values independently from the set $\{x_1, x_2, \cdots, x_N\}$, it follows from the above mapping that each w_i also takes up N possible values independently from the set $\{x_1, x_2, \cdots, x_N\}$. Hence, the set

$$\Big\{ \sum_{i=1}^{q} w_i - \sum_{k=q+1}^{2q} w_k, w_i \in \{x_1, \cdots, x_N\} \forall i \Big\}$$

represents the $2q$th order difference co-array of the physical array, which is independent of the orientation indexed by l. So, the elements of $\mathbf{a}_{virtual}^{2q}(\theta_d)$ correspond to this new virtual array, which is the same irrespective of the orientation l. Defining

$$\mathbf{B}_{2q}(\boldsymbol{\theta}) \triangleq [\mathbf{a}_{virtual}^{2q}(\theta_1) \quad \mathbf{a}_{virtual}^{2q}(\theta_2) \cdots \mathbf{a}_{virtual}^{2q}(\theta_D)]$$

we can rewrite (4.14) in matrix notation to obtain (4.12). Here the dth column of $\mathbf{B}_{2q}(\theta)$ corresponds to $\mathbf{a}_{virtual}^{2q}(\theta_d)$ which is the steering vector corresponding to the $2q$th order difference co-array and hence $\mathbf{B}_{2q}(\theta)$ behaves like the manifold of the $2q$th order virtual co-array. □

4.4.3 Independence from Orientations

It can be seen from the above theorem that the entries in $\mathbf{C}_{2q,\mathbf{y}}(l)$ inherently contain the $2q$th order difference set and on vectorizing, it behaves like the model of signals received by the $2q$th order difference co-array. In particular, it also shows that the cumulant matrices corresponding to all the different arrangements given in [44] give rise to the same $2q$th order difference co-array, hence making the model in (4.12) independent of any particular arrangement of the $\mathbf{C}_{2q,\mathbf{y}}(l)$. Let $S(l)$ denote the set of virtual sensor positions associated with the virtual steering vector $\mathbf{a}_{virtual}^{(l)}(\theta)$. Then we have the following corollary:

Corollary 4.4.1. *The 2nd order difference set (or the difference co-array) of $S(l)$ is the same for each l and it is equal to the $2q$th order difference co-array of the physical array.* □

Proof. The entries of $\mathbf{a}_{virtual}^{(l)*}(\theta_d) \otimes \mathbf{a}_{virtual}^{(l)}(\theta_d)$ contain the 2nd order difference co-array of $S(l)$ and from (4.16), we find that they also represent the $2q$th order difference co-array of the original physical array, which is only a function of the sensor positions in the original array. Hence the 2nd order difference co-array of $S(l)$ is identical for all l. □

Hence, although the virtual arrays associated with $\mathbf{a}_{virtual}^{(l)}(\theta)$ can be very different for different l, their difference co-arrays however are identical.We illustrate this with the following example:

Example: Figure 4.1 (a) shows an arbitrarily placed 3 sensor array in two dimensions. If we compute the 4th order cumulants of signal received by this array, then the cumulant matrix can be arranged in two ways indexed by l where $1 \leq l \leq q = 2$ which gives rise to two different virtual arrays. For $l = 1$, the virtual array has its elements located at $\mathbf{x}_i - \mathbf{x}_j, \quad 1 \leq i, j \leq 3$ where $\mathbf{x}_i$ denotes the location of the physical array sensors. This denotes the so-called difference co-array. Figure 4.1 (b) shows this array. Whereas for $l = 2$, the virtual array has its elements located at $\mathbf{x}_i + \mathbf{x}_j, \quad 1 \leq i, j \leq 3$ which is shown in Fig. 4.1 (c). This denotes the so called sum co-array. Notice that these two virtual arrays look quite different which justifies the claim in [44] that each arrangement of the cumulant matrix gives rise to different processing powers. However, the difference co-array of both these arrays are *identical* and it corresponds to the 4th order difference

co-array of the physical array. This array is shown in Fig. 4.1 (d) and it has a lot more elements than the individual arrays. This array is captured by the model in (4.12).

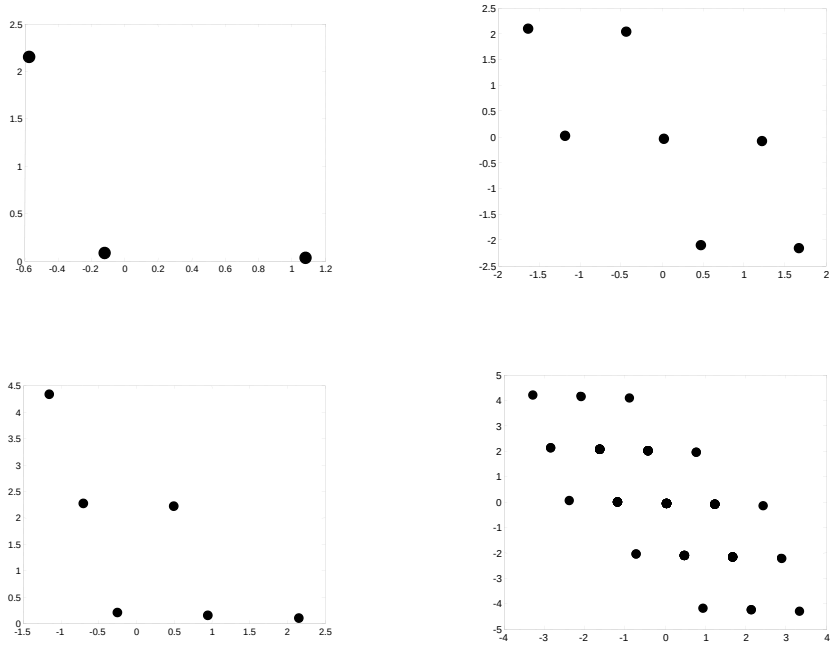


Figure 4.1: An arbitrary physical array and its various higher order virtual arrays (top left) The physical array with 3 arbitrarily generated sensor locations (top right) Virtual array (difference co-array) for $l = 1, q = 2$. (bottom left) Virtual array (sum co-array) for $l = 2, q = 2$. (bottom right) $2q = 4$th order difference co-array.

When subspace based algorithms were applied on $\mathbf{C}_{2q,\mathbf{y}}(l)$, the degrees of freedom obtained were heavily dependent on the choice of l [44] since each l can potentially give rise to a different virtual array. However, in the subsequent sections, we will consider the model in (4.12) and develop a new DOA estimation algorithm which will exploit the degrees of freedom of the $2q$th order difference co-array, which is independent of the particular arrangement given by l. As will be seen, the proposed DOA estimation method works for special classes of arrays, whose $2q$th order difference co-array contains an ULA segment. The identifiability properties of the proposed method are equuivalent to the identifiability properties of the ULA segment. Though $2q$ MUSIC does not require the physical array to possess any such property, if the virtual array exploited by the $2q$ MUSIC algorithm is not an ULA, ambiguity in DOA estimation will arise [3]. So, to ensure unambiguous $1D$ DOA estimates, it is necessary for the virtual array seen by the $2q$ MUSIC algorithm to be a

ULA.

4.5 Nested Array with $2q$ levels and its virtual array

In this section, we propose a family of non uniform arrays, (first introduced in [135]), called the $2q$ level nested array, parameterized by q, which can give rise to $O(N^{2q})$ virtual sensors in the $2q$th order difference co-array. Specifically, we will see that for each q, the $2q$ level nested array with N physical sensors can be systematically constructed as a combination of $2q$ uniform linear arrays with appropriate spacing and its $2q$th order difference co-array will contain an ULA with $O(N^{2q})$ virtual sensors.

Definition 4.2: (**Nested Array with $2q$ levels**:) A $2q$ level nested array (characterized by integers q, N_1, N_2,$\cdots N_{2q}$) is a non uniform array consisting of $2q$ uniform linear arrays with a total of $N = \sum_{i=1}^{2q}(N_i - 1) + 1$ sensors, where the ith ULA ($1 \leq i \leq 2q-1$) has $N_i - 1$ sensors located at

$$\{n(\prod_{k=0}^{i-1} N_k)\lambda/2, \quad n = 1, 2, \cdots, N_i - 1\}$$

with the convention of $N_0 = 1$, and the $2q$th level has N_{2q} sensors located at

$$\{n(\prod_{k=0}^{2q-1} N_k)\lambda/2, \quad n = 1, 2, \cdots, N_{2q}\}$$

□

As can be seen, the ULAs for higher levels are increasingly sparser, with the spacing having a direct relation to the number of sensors in the ULAs in the previous levels. The increased interelement spacing in each level gets filled up with the sensors in the denser ULAs as one considers successively higher order difference co-arrays, giving rise to a ULA with $O(\prod_{i=1}^{2q} N_i)$ sensors for the $2q$th order difference co-array for a $2q$ level nested array, although the physical array has only $N = \sum_{i=1}^{2q} N_i - 2q + 1$ sensors. This is proved formally in the following theorem:

Theorem 4.5.1. ***($2q$th order Difference Co-Array of $2q$ level Nested Array)*** *The $2q$th order difference co-array of a $2q$ level nested array contains a uniform linear array with*

$$2\prod_{i}^{2q} N_i - 1$$

virtual sensors whose locations are given by

$$\{\pm n\lambda/2, n = 0, 1, \cdots, \prod_{i=1}^{2q} N_i - 1\}.$$

□

Proof. See Appendix 4.11.1.1. □

The following remarks hold for the $2q$ order nested array:

1. The proposed $2q$ order nested array is a generic family which defines a class of arrays, one for each q, which gives rise to $O(N^{2q})$ virtual sensors *continuously* in the form of a standard ULA with spacing $\lambda/2$. While there exist examples of arrays for $q = 1$ (MRA, 2 level nested array etc.) whose *second* order difference co-array is a ULA with $O(N^2)$ sensors, this class of linear arrays is the first of its kind which works for each $q \geq 2$.

2. The $2q$ level nested array was briefly introduced in [135] in the context of second order difference co-array. However, for $q \geq 2$, the second order difference co-array of a $2q$ level nested array is *not* a ULA but it has holes (missing sensors), which does not render it suitable for DOA estimation using the method proposed in [135]. However, it produces a filled ULA when the $2q$th order difference co-array is considered, which makes it a very appropriate choice for DOA estimation using $2q$th order cumulants.

4.5.1 Bonus ULA segment

For $q \geq 2$, an interesting property of the $2q$th order difference co-array of a $2q$ level nested array is that, the ULA segment is actually bigger than the main ULA segment given in Theorem 4.5.1. The extra or bonus segment extends out on both sides of the main ULA segment and contains $\prod_{i=1}^{2q-1} N_i$ sensors on each side. Hence, it adds to the total number of contiguous virtual sensors, as given by the following lemma:

Lemma 4.5.1. *The $2q$th order difference co-array of a $2q$ level nested array contains an ULA with*

$$2(\prod_{i=1}^{2q} N_i + \prod_{i=1}^{2q-1} N_i) - 1$$

elements located at

$$\{\pm n\lambda/2, \quad -L+1 \leq n \leq L-1, \quad L = \prod_{i=1}^{2q} N_i + \prod_{i=1}^{2q-1} N_i\}$$

□

Proof. See Appendix 4.11.1.2. □

4.5.2 Comments on the main and bonus ULA segments

Some comments follow regarding the main and bonus ULA segments of the $2q$ level nested array:

- The bonus ULA segment does not occur for a 2 level nested array (i.e., corresponding to $q = 1$). It appears for $q = 2$ and through a recursive argument (see Appendix 4.11.1-B), it applies to all values of $q > 2$.
- For $q \geq 2$, the $2q$th order difference co-array of the $2q$ level nested array contains a continuous ULA segment which is the union of the main and bonus segments.
- It will be shown in the next section that in the optimal $2q$ level nested array, the main ULA segment contains $O(N^{2q})$ elements whereas the contribution of the extra ULA segment is $O(N^{2q-1})$. For small N, the contribution of the extra segment is significant and it becomes proportionately smaller for large N.
- Fig. 4.2 illustrates the main and bonus ULA segments of a 4 level nested array with $N_1 = 3$, $N_2 = N_3 = N_4 = 2$. The main ULA segment extends from -23 to 23 where $N_1N_2N_3N_4 - 1 = 23$. Also, there is an extra ULA segment extending out on each side of this main ULA segment and it has $N_1N_2N_3 = 3 \times 2 \times 2 = 12$ elements on either side.

4.6 Optimization of the number of virtual sensors

In this section, we will address the problem of maximizing the degrees of freedom in the ULA contained in the $2q$ level nested array under the constraint of a fixed number of sensors. First we will solve for the optimal sensor distribution which maximizes the virtual sensors in the main ULA. We will then derive the solution which maximizes the total number of elements in both the main ULA and the adjoining extra ULA segments.

4.6.1 Optimal Sensor Allocation Problem (without the extra segment)

According to Theorem 4.5.1, the $2q$th order difference co-array of the $2q$ level nested array contains an ULA with $2\prod_{i=1}^{2q} N_i - 1$ elements (without considering the extra ULA segment). Hence, the degrees of freedom optimization problem can be formulated as

$$\max_{N_1,\cdots,N_{2q}\in\mathbb{N}^+} (\prod_{i=1}^{2q} N_i) \tag{4.17}$$

subject to $\sum_{i=1}^{2q} N_i = N + 2q - 1$.

In other words, we are interested in how to distribute the total fixed (N) number of sensors between the $2q$ levels which gives rise to a virtual ULA with largest possible number of elements. If we apply the Arithmetic Mean-Geometric Mean ($AM - GM$) inequality to (4.17), we find that $\prod_{i=1}^{2q} N_i$ is maximized when

$$N_i = \frac{N + 2q - 1}{2q}, \quad i = 1, 2, \cdots, 2q.$$

However this is not a valid integer solution unless $N + 2q - 1$ is a multiple of $2q$. Hence, to solve (4.17) for all possible values of $N + 2q - 1$, let us first define integers m and n to be the quotient and remainder of $N + 2q - 1$ modulo $2q$, i.e.,

$$N + 2q - 1 = 2qm + n, \quad 0 \leq n \leq 2q - 1$$

The solution to (4.17) can then be completely characterized in terms of m and n as given by the following theorem. Note that since the problem is permutation symmetric, if $\{N_i^{(opt)}\}$ is a solution, then so is any permutation.

Theorem 4.6.1. *One solution to the optimization problem in (4.17) is given by* $\{N_i^{(opt)}\}$ *where* $N_i^{(opt)} = m + 1, \quad i = 1, 2, \cdots, n,$
$N_i^{(opt)} = m, \quad i = n + 1, n + 2, \cdots, 2q$ *The corresponding optimal number of elements in the main ULA segment in the* $2q$*th order difference co-array of the* $2q$ *level nested array is*

$$2m^{2q-n}(m + 1)^n - 1$$

$\diamondsuit$ $\square$

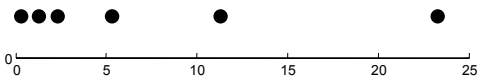
Proof. See Appendix 4.11.1.2. □

Hence, we find that when $N+2q-1$ happens to be a multiple of $2q$ we get the virtual ULA with a total of

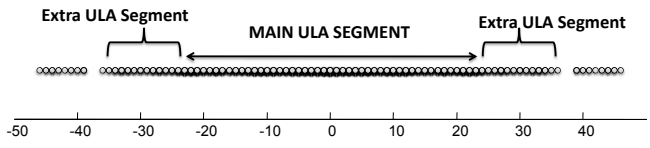
$$2\Big(\frac{N+2q-1}{2q}\Big)^{2q}-1$$

sensors. For other values of N, the number of virtual sensors is also always of $O(N^{2q})$. Hence the $2q$ level nested array gives proportionately increasing degrees of freedom with increasing q. Hence this provides a very generic structure for each higher level difference co-array where virtual array grows as the $2q$th power of N as N increases. To the best of our knowledge, this has not been reported before in the past.

Example: Figure 4.2 shows an example of optimal 4 level nested array with $N=6$ physical sensors. Here $m=2, n=1$. The choice of $N_1=3$, $N_2=N_3=N_4=2$ corresponds to an optimal solution which is shown. The corresponding 4th order difference coarray contains the optimal number $2m^{2q-1}(m+1)-1=2\times 8\times 3-1=47$ of sensors in the form of a ULA from -23 to 23 as shown in the figure. Also, there is an extra ULA segment extending out on each side of this main ULA segment and it has $N_1N_2N_3=3\times 2\times 2=12$ elements on either side. Hence, starting with 6 physical sensors in the 4 level nested array, we get a total of $47+2\times 12=71$ virtual elements in the 4th order difference co-array.



(a) Optimal 4 Level Nested array with N=6 sensors.



(b) Corresponding 4th order Difference Co-Array

Figure 4.2: Optimal 4 level nested array (optimized without considering the extra ULA segment) with 6 sensors and its 4th order difference co-array. Notice the extra ULA segments that extend out.

4.6.2 Relation to exponential array

There exists an interesting result when the number of sensors (N) is equal to $2q+1$, as given by the following Corollary to Theorem 4.6.1:

Corollary 4.6.1. *The exponential array with sensor locations*

$$\{2^{n-1}\lambda/2, \quad n = 1, 2, \cdots 2q+1\}$$

is the optimal $2q$ level nested array with $2q+1$ sensors, i.e., it is the solution to (4.17) when $N = 2q+1$. □

Proof. Follows directly as a corollary of Theorem 4.6.1. □

The exponential array was also obtained in [135] as the result of a different optimization problem. It was obtained when the degrees of the freedom in the *second order difference set* of the K-level nested array was optimized, with respect to K and the number of sensors $N_k, k = 1, 2, \cdots K$ in the K levels (see Theorem 1 and Corollary 1 in [135]). Though the exponential array could achieve the maximum number of virtual sensors possible ($N(N-1)+1$) in the second order difference set of any array with N sensors, the second order difference co-array was *not* a filled ULA (it had holes), thus rendering it inappropriate for DOA estimation with second order statistics using the algorithm proposed in [135]. However, as we proved here, the $2q+1$ element exponential array is the optimal array when $2q$th order difference sets are considered and it gives rise to a ULA with $2(2^{2q})-1$ elements, which can be readily used for DOA estimation, as shown in Sec 4.7.

4.6.3 Optimization including the extra ULA segment

If we include the extra ULA segment as given in Lemma 4.5.1, then the optimization problem becomes

$$\max_{N_1,\cdots,N_{2q}\in\mathbb{N}^+} \left(\prod_{i=1}^{2q} N_i + \prod_{i=1}^{2q-1} N_i\right) \tag{4.18}$$

subject to $\sum_{i=1}^{2q} N_i = N + 2q - 1$

The solution to this problem is given by the following Corollary to Theorem 4.6.1:

Corollary 4.6.2. *One solution to (4.18) is given by* $N_i = m_1 + 1, \quad i = 1, 2, \cdots, n_1,$

$N_i = m_1, \quad n_1 + 1 \leq i \leq 2q-1$

$N_i = m_1 - 1, \quad i = 2q$ *Here* m_1 *and* n_1 *are the quotient and remainder respectively of* $N + 2q$ *modulo* $2q$*. The corresponding maximized number of elements in the ULA segment of the* $2q$*th order difference co-array of the* $2q$ *level nested array is* $2m_1^{2q-n_1}(m_1+1)^{n_1}$*.* □

Proof. Defining $\mathrm{N}_i^{'} = N_i, \quad 1 \leq i \leq 2q-1$
$N_{2q}^{'} = N_{2q} + 1$ we can rewrite the problem in (4.18) as

$$\max_{N_1^{'},\cdots,N_{2q}^{'} \in \mathbb{N}^+} (\prod_{i=1}^{2q} N_i^{'})$$

subject to $\sum_{i=1}^{2q} N_i^{'} = N + 2q$
Then the problem (4.18) is identical to the problem in (4.17), the only exception being that the right hand side of the constraint has $N+2q$ instead of $N+2q-1$. Hence, as a corollary to Theorem 4.6.1, the optimal solution set $\{N_i^{'}\}$ is of the form $\mathrm{N}_i^{'} = m_1 + 1, \quad 1 \leq i \leq n_1,$
$N_i^{'} = m_1, \quad n_1 + 1 \leq i \leq 2q$ where m_1 and n_1 are the quotient and remainder respectively of $N+2q$ modulo $2q$. □

Let $N + 2q - 1 = 2qm + n$ and we fix N. Then, if $n \neq 2q-1$, $m_1 = m$ and $n_1 = n+1$, otherwise, $m_1 = m+1$,$n_1 = n$. In the first case, the maximized number of virtual elements in (4.18) is $2m^{2q-n-1}(m+1)^{n+1} - 1$ whereas in the second case, it is $2(m+1)^{2q-n}(m+2)^n - 1$. The maximized value for the problem in (4.17) however is $2(m+1)^n m^{2q-n} - 1$. Taking the difference, it can be easily seen that in the optimal case, the extra ULA segments add $O(m^{2q-1})$ (i.e., $O(N^{2q-1})$) more virtual sensors to the main ULA segment containing $O(N^{2q})$ elements.

4.7 Underdetermined DOA estimation with $2q$ level Nested Array

In this section, we will demonstrate how to perform DOA estimation of $O(N^{2q})$ sources from $2q$th order cumulants of the received data, using an extension of the DOA estimation algorithm originally proposed in [135] for the special case of $q = 1$. The method in [44] could identify only upto $O(N^q)$ sources and used a circular array to demonstrate it. In $1D$, no systematic class of arrays has so far been proposed or used by $2q$ MUSIC which can identify $O(N^q)$ angles. However, in this chapter, $O(N^{2q})$ sources are shown to be identifiable with the $2q$ level nested array using the

algorithm proposed in this section.

4.7.1 Proposed Spatial Smoothing Based Algorithm

Let us consider the model in (4.12) where the matrix $\mathbf{B}_{2q}(\theta)$ contains a $(2M_{2q}+1)\times D$ Vandermonde submatrix $\mathbf{A}_{2q}^{virtual}(\theta)$ corresponding to the ULA in the $2q$th order difference co-array. Here M_{2q} denotes the degrees of freedom on positive side of the virtual ULA. For example $M_{2q} = ((N+2q-1)/2q)^{2q}-1$ for the case when $N+2q-1$ is a multiple of $2q$. Note that we can partition

$$\mathbf{A}_{2q}^{virtual}(\theta) = \begin{pmatrix} \mathbf{U}_{2q}^{virtual}(\theta) \\ \mathbf{L}_{2q}^{virtual}(\theta) \end{pmatrix} \tag{4.19}$$

where the elements of $\mathbf{U}_{2q}^{virtual}(\theta) \in \mathcal{C}^{(M_{2q}+1)\times D}$ and $\mathbf{L}_{2q}^{virtual}(\theta) \in \mathcal{C}^{(M_{2q})\times D}$ are given respectively by $[\mathrm{U}_{2q}^{virtual}(\theta)]_{m+M_{2q}+1,n} = e^{j\pi m \sin\theta_n}$,
$-M_{2q} \le m \le 0, 1 \le n \le D$

$$[\mathbf{L}_{2q}^{virtual}(\theta)]_{m,n} = e^{j\pi m \sin\theta_n}, 1 \le m \le M_{2q}, 1 \le n \le D$$

By extracting the appropriate rows from $\mathbf{c}_{vec}$ in (4.12), we get $\mathbf{z}_1 \in \mathcal{C}^{(2M_{2q}+1)\times 1}$ where

$$\mathbf{z}_1 = \mathbf{A}_{2q}^{virtual}(\theta)\mathbf{p} \tag{4.20}$$

Comparing with (4.1), we find that this behaves like the received signal model of a $(2M_{2q}+1)$ element ULA with the signal amplitudes being replaced by the $2q$th order cumulants of the signals (elements of $\mathbf{p}$). Also the noise being Gaussian, its $2q$th order cumulant ($q \ge 2$) is zero, which explains the absence of a noise term. To apply subspace based methods for DOA estimation using this model, would require us to build a "covariance-like" matrix with the appropriate rank of null space. However in the model in (4.20), the source vector $\mathbf{p}$ behaves like perfectly correlated sources. This leads us to take inspiration from the concept of spatial smoothing [74],[163] for building up a covariance like matrix where the sources will be decorrelated by spatial averaging. This is a modification of our spatial smoothing based approach proposed in [135] because the model in (4.20) is more simplistic due to absence of a noise term. The algorithm consists of the following

steps:

1. Divide the vectors $\mathbf{z}_1$ into subvectors $\mathbf{z}_{1i} \in \mathcal{C}^{(M_{2q}+1)\times 1}, i = 1, 2, \cdots, M_{2q}+1$ where

$$[\mathbf{z}_{1i}]_m = [\mathbf{z}_1]_{i+m-1}.$$

2. Construct the positive semidefinite matrix $\mathbf{C}_{virtual} \in \mathcal{C}^{(M_{2q}+1)\times(M_{2q}+1)}$ as

$$\mathbf{C}_{virtual} = \frac{1}{M_{2q}+1} \sum_{i=1}^{M_{2q}+1} \mathbf{z}_{1i}\mathbf{z}_{1i}^H$$

3. Apply subspace based algorithms like MUSIC on $\mathbf{C}_{virtual}$ to perform DOA estimation.

The complete algorithm is summarized in Table 4.1.

The identifiability of the proposed algorithm is given by the following theorem:

Theorem 4.7.1. *The matrix $\mathbf{C}_{virtual}$ can be expressed as*

$$\mathbf{C}_{virtual} = \frac{1}{M_{2q}+1}\mathbf{C}\mathbf{C}^H$$

$$\mathbf{C} = \mathbf{U}_{2q}^{virtual}(\theta)\mathbf{\Lambda}\mathbf{U}_{2q}^{virtual}(\theta)^H$$

where $\mathbf{\Lambda} \in \mathcal{R}^{D\times D}$ is a diagonal matrix with $\mathbf{\Lambda}_{i,i} = \mathbf{p}_i, i = 1, 2, \cdots, D$ and $\mathbf{U}_{2q}^{virtual}$ is given in (4.19). Application of subspace based DOA estimation algorithms on $\mathbf{C}_{virtual}$ can identify upto M_{2q} source directions. □

Proof. Notice that

$$\mathbf{z}_{1i} = \mathbf{U}_{2q}^{virtual}(\theta)\mathbf{\Phi}^{i-1}\mathbf{p}$$

where $\mathbf{\Phi}(\theta)$ is a diagonal matrix whose diagonal elements are given by

$$[\mathbf{\Phi}]_{i,i} = e^{j\pi \sin\theta_i}, i = 1, 2, \cdots, D.$$

Then, $(\mathrm{M}_{2q}+1)\mathbf{C}_{virtual}$
$= \mathbf{U}_{2q}^{virtual}(\theta)\Big(\sum_{i=1}^{M_{2q}+1} \mathbf{\Phi}^{i-1}\mathbf{p}\mathbf{p}^H\mathbf{\Phi}^{(i-1)H}\Big)\mathbf{U}_{2q}^{virtual}(\theta)^H$

$= \mathbf{U}_{2q}^{virtual}(\theta)\mathbf{\Lambda}\Big(\sum_{i=1}^{M_{2q}+1}\boldsymbol{\phi}_{i-1}\boldsymbol{\phi}_{i-1}^{H}\Big)\mathbf{\Lambda}^{H}\mathbf{U}_{2q}^{virtual}(\theta)^{H}$
$= \mathbf{U}_{2q}^{virtual}(\theta)\mathbf{\Lambda}\mathbf{U}_{2q}^{virtual}(\theta)^{H}\mathbf{U}_{2q}^{virtual}(\theta)\mathbf{\Lambda}^{H}\mathbf{U}_{2q}^{virtual}(\theta)^{H}$ where $\boldsymbol{\phi}_i = \mathcal{C}^{D\times 1}$ is given by

$$[\boldsymbol{\phi}_i]_k = [\mathbf{\Phi}^i]_{k,k}, \quad 1 \leq k \leq D, 0 \leq i \leq M_{2q}$$

and the last equality in (4.7.1) is due to the fact that

$$\sum_{i=1}^{M_{2q}+1}\boldsymbol{\phi}_{i-1}\boldsymbol{\phi}_{i-1}^{H} = \mathbf{U}_{2q}^{virtual}(\theta)^{H}\mathbf{U}_{2q}^{virtual}(\theta)$$

Since $\mathbf{U}_{2q}^{virtual}(\theta)$ is a Vandermonde matrix of rank D, $\mathbf{C}_{virtual}$ is also of rank D and its null space (of dimension $M_{2q}+1-D$), spanned by the eigenvectors corresponding to the zero eigenvalues, is the same as the null space of $\mathbf{U}_{2q}^{(virtual)}(\theta)^{H}$. Hence, by performing an eigenvalue decomposition of $\mathbf{C}_{virtual}$ and using subspace based methods like MUSIC to obtain the pseudo spectrum, upto $D \leq M_{2q}$ sources can be uniquely identified. □

Notice that, similar to ULA based DOA estimation techniques, uniqueness (no false peaks) is guaranteed due to the ULA structure of the virtual array which forces $\mathbf{U}_{2q}^{virtual}(\theta)$ to have the Vandermonde property.

4.7.2 Comparison with $2q$ MUSIC

Prior to the algorithm proposed in this chapter, the established algorithm for DOA estimation using $2q$th order cumulants was the $2q$ MUSIC algorithm and its variants [44],[14]. Comparison between the proposed algorithm and the $2q$ MUSIC algorithm for $1D$ DOA estimation problem, in terms of required array geometry, identifiability, numerical complexity, and ability to handle correlated sources is tabulated in Table 4.2.

Though the $2q$ MUSIC algorithm is applicable for non ULA virtual arrays, it might suffer from a fundamental ambiguity problem that occurs for subspace based DOA estimation algorithms applied on non ULAs [3]. In other words, two different angles can map to the same steering vector, resulting in ambiguity. However, since the proposed algorithm always exploits a ULA structure, such ambiguity does not arise.

The major computational complexity arises from the singular value decomposition (SVD) of the cumulant matrix (for $2q$ MUSIC) or the proposed covariance-like matrix, which is the first step

towards application of subspace based DOA estimation techniques. For a $N \times N$ matrix, the complexity of SVD is $O(N^3)$. Since the matrix under consideration for $2q$ MUSIC is at most of size $O(N^q) \times O(N^q)$ and for the proposed method, it is at most of size $O(N^{2q}) \times O(N^{2q})$, the respective complexities become $O(N^{3q})$ and $O(N^{6q})$.

Hence, the proposed algorithm is capable of identifying larger number of statistically independent sources compared to the $2q$ MUSIC. On the other hand, $2q$ MUSIC can handle correlated sources and also requires less computations. Further comparisons between the performance of these two algorithms in terms of variation of RMSE with respect to SNR, snapshots, modeling errors and resolution are numerically performed in Sec 4.9.

4.8 Comments on Identifiability and Cramer Rao Bound

In this section, we discuss the maximum number of sources that can be identified by *any method* that uses the $2q$th order cumulants of the received data. We relate the identifiability of the proposed method to this bound. We also address the important issue of Cramer Rao bound analysis [169],[81],[168] for our model for the particular case of number of sources exceeding the number of physical sensors.

4.8.1 Identifiability

Among DOA estimation methods using the $2q$th order cumulants of the received data, the previously reported method named $2q$ MUSIC could identify at most $O(N^q)$ statistically independent sources. The method proposed in this chapter can identify a number of sources which is equal to half of the number of virtual elements in the $2q$th order difference co-array of the physical array. This can be as large as $O(N^{2q})$ statistically independent sources for appropriately designed arrays. This brings us to the question: "Fundamentally, what is the maximum number of statistically independent sources that can be identified by any method using the $2q$th order cumulants of the received data?" The question of fundamental identifiability in DOA estimation problems using sensor arrays has previously been addressed in [20],[90],[197],[6],[3]. However, they either consider a Gaussian model (for which higher order cumulants are zero) or consider the source signals to be deterministic. In this chapter, for the first time, we attempt to answer the question of fundamental identifiability of any method that exploits $2q$th order cumulants of the received data, for any $q \geq 1$.

We first provide a necessary condition that provides an upper bound to this number.

4.8.2 An Upper Bound on Identifiability

The upper bound on the number of identifiable sources for $2q$th order statistic based methods stem from the "Implicit function theorem" and it is also known as "Equations vs Unknowns" bound. This bound has been used in the past to prove results on identifiability of DOA parameters for Gaussian source models (see [197],[6] and the references therein). The bound states: *"The number of real valued unknown parameters should be less or equal to the number of independent equations (extracted from the $2q$th order cumulant values in our case)"*. We assume knowledge of perfect cumulant values, which is true in the limit of the number of snapshots tending to infinity.

- **Number of Real Unknowns:**We have a total of N^{2q} cumulant values (possibly repeated) which are given by (4.14), (4.16) $\text{Cum}(\text{n}_1, \cdots, n_{2q}) = \sum_{d=1}^{D} c_{2q,s_d} e^{j\frac{2\pi}{\lambda}(\sum_{i=1}^{q} x_{n_i} - \sum_{i=q+1}^{2q} x_{n_i})\sin\theta_d}$, $1 \leq n_i \leq N$ where x_n denotes the physical location of the nth sensor. There are a total of $2D$ real unknown parameters, viz.,
  (1) D source DOA parameters: $\{\theta_d, \quad 1 \leq d \leq D\}$, and
  (2) D $2q$th order self cumulant values of the D sources: $\{c_{2q,s_d}, \quad 1 \leq d \leq D\}$

- **Number of Independent equations:** Each of the N^{2q} cumulant values in (4.8.2) gives rise to an equation in the above $2D$ unknowns. However, not all N^{2q} equations are independent. We can see from (4.8.2) that the number of *distinct* equations is equal to the number of distinct values in the set $\{\sum_{i=1}^{q} x_{n_i} - \sum_{i=q+1}^{2q} x_{n_i}, 1 \leq n_i \leq N\}$ which is nothing but the $2q$th order difference set of the physical array. Denoting $N_{2q}^{(ca)}$ as the number of elements in this set, the number of distinct equations therefore becomes $N_{2q}^{(ca)}$.

According to the equations v/s unknowns bound, for reliable identifiability, we should have [6]:
2D =No. of Unknowns
$\leq$ No. of Independent Equations
$\leq$ No. of Distinct Equations
$= N_{2q}^{(ca)}$. Therefore we get an upper bound on the maximum number of identifiable sources as $D \leq \lfloor \frac{N_{2q}^{(ca)}}{2} \rfloor$ Since the difference set is always symmetric about zero and it includes the zero, $N_{2q}^{(ca)}$

is always odd and hence we can modify the upper bound as

$$D \leq \frac{N_{2q}^{(ca)} - 1}{2} \tag{4.21}$$

4.8.3 Achieving the Upper Bound with the Proposed Method

We now consider the special class of arrays whose $2q$th order difference set is a uniform linear array. Let us call this class as "$2q$ ULA". The 2 level nested array is a $2q$ ULA, corresponding to $q = 1$. For such arrays, the proposed spatial smoothing based algorithm effectively builds a covariance matrix using $M_{2q} + 1$ elements of the virtual array where $N_{2q}^{(ca)} = 2M_{2q} + 1$. Subsequently, subspace based algorithms acting on this covariance matrix can identify any D DOA angles provided

$$D \leq M_{2q} = \frac{N_{2q}^{(ca)} - 1}{2}.$$

Hence for the class of "$2q$ ULA" arrays, the proposed algorithm can indeed achieve the upper bound given by (4.21). This leads us to the following theorem:

Theorem 4.8.1. *In $1D$ DOA estimation problems involving D statistically independent sources and using $2q$th order cumulants of the received data, the number of identifiable sources D, is always bounded by*

$$D \leq \frac{N_{2q}^{(ca)} - 1}{2}$$

where $N_{2q}^{(ca)}$ denotes the cardinality of the $2q$th order difference set of the physical sensor locations. Moreover, if the physical array is such that the $2q$th order difference co-array is a ULA, the proposed spatial smoothing based algorithm can achieve this bound, i.e., it can identify upto $\frac{N_{2q}^{(ca)}-1}{2}$ source DOAs unambiguously. □

The following comments follow from the above theorem:

1. Although the proposed spatial smoothing based algorithm uses only about half of the $2q$th order difference set, this does not result in suboptimal identifiability (i.e., we do not really "lose" half of the freedom) because it achieves the maximum identifiability we can obtain.

2. For a uniform linear array, $N_{2q}^{(ca)} = 2q(N-1)+1$ and hence the maximum number of identifiable sources using $2q$th order cumulants is given by $D \leq q(N-1)$. The traditional $2q$ MUSIC [44] can in fact attain this (even without using the $2q$th order difference set). So the proposed method does not provide increased identifiability if a physical ULA is used.

3. The maximum value of $N_{2q}^{(ca)}$ is $O(N^{2q})$ and the $2q$ level nested array's $2q$th order difference set contains $O(N^{2q})$ elements. Hence, the proposed method acting on the $2q$th order nested array can identify $O(N^{2q})$ sources.

4.8.4 Cramer Rao Bound Study

There are traditionally two types of Cramer Rao bound that are identified, depending on the statistical characterization of the source signals. Consider the model in (4.1). We can consider two separate Cramer Rao bounds [168] as follows:

1. **Deterministic Cramer Rao Bound**[168],[169]: If the source signals $\{s_d(t), 1 \leq d \leq D\}$ are considered as deterministic (non random) unknown waveforms, the corresponding Cramer Rao bound is known as *deterministic or conditional Cramer Rao bound*.

2. **Stochastic Cramer Rao Bound**[168],[81]: If the source signals $\{s_d(t), 1 \leq d \leq D\}$ are random (i.e., for each d, the sequence $\{s_d(t)\}_{t=1}^{t=T}$, varies from realization to realization), then the corresponding Cramer Rao bound is known as the *Stochastic or Unconditional Cramer Rao bound*. Evidently, in this case, the Cramer Rao bound depends on the probability distribution of the random source signal waveforms.

In our case, we are interested in the underdetermined scenario (i.e., $D > N$). It is to be noted that in our model, we assume the source signals to be random and we perform DOA estimation by computing the cumulants of their distribution. So, the deterministic model loses its relevance for our model. However, we will show that even if we had treated the source signal waveforms to be unknown deterministic quantities, there would exist no consistent estimator for the DOAs by the following argument.

4.8.4.1 Deterministic Cramer Rao Bound

The deterministic CRB for the model under consideration was rigorously derived and studied in [169]. However the final closed form expression for the deterministic CRB obtained in [169] only applies to the case $D \leq N$. We will now show that the when $D > N$, *the Fisher Information matrix for the deterministic model is singular*, as follows:

- **Singularity of Fisher Information Matrix:** Equation (E.9) in Appendix E of [169] provides the structure of the deterministic FIM. Denoting $\mathbf{\Theta} = [\omega_1 \cdots \omega_D]^T$ where $\omega_i = \pi \sin\theta_i$, the

CRB Covariance matrix, $\mathbf{CRB}(\mathbf{\Theta})$ for the estimate of $\mathbf{\Theta}$, corresponds to the lower rightmost $D \times D$ block of $\mathbf{\Sigma}$. In [169], the expression for $\mathbf{CRB}(\mathbf{\Theta})$ is derived under the assumption $D < N$. For $D > N$ (which is the primary focus of this chapter), the FIM actually turns out to be singular and hence cannot be inverted. This is because the array manifold matrix $\mathbf{A}(\mathbf{\Theta})$ is a $N \times D$ matrix and hence the matrix

$$\mathbf{H} = \mathbf{A}(\mathbf{\Theta})^H \mathbf{A}(\mathbf{\Theta})$$

becomes rank deficient matrix for $D > N$. So each block $\begin{pmatrix} \bar{\mathbf{H}} & -\tilde{\mathbf{H}} \\ \tilde{\mathbf{H}} & \bar{\mathbf{H}} \end{pmatrix}$ on the diagonal of the FIM in Equation (E.9) in [169] becomes rank deficient as well. Here $\bar{\mathbf{H}}$ and $\tilde{\mathbf{H}}$ denote the real and imaginary parts of $\mathbf{H}$ respectively. So the FIM becomes singular.

- **Implications of Singular Fisher Information Matrix:** The implications of the singular Fisher Information matrix has been addressed in depth in [167]. In such cases, the Cramer Rao bound is obtained by considering the Moore-Penrose pseudoinverse of the FIM. However, as pointed out in [167], such a Cramer Rao bound is misleading because when the FIM is singular, there exists no unbiased estimator of the parameters with finite variance, except under unusual conditions. So in such situations, the unknown parameters cannot be reliably estimated.

In the light of the above discussion, the deterministic Cramer Rao bound loses its relevance for the underdetermined scenario. We now examine the stochastic CRB which is the appropriate tool for the model under consideration.

4.8.4.2 Stohcastic Cramer Rao Bound

Derivation of the stochastic Cramer Rao bound is more challenging than the deterministic counterpart and it has been discussed in [81]. The stochastic CRB depends on the type of the distribution that the random source signals $\mathbf{s}(t)$ are assumed to follow. In DOA estimation problems, usually the source signals are assumed to be Gaussian for which the CRB has been studied in [81], [3]. However, the Gaussian model is not appropriate for our case since for the Gaussian model, the $2q$th order cumulants, for $q \geq 2$, are zero. In the simulations of this chapter, we assume the signals to be BPSK modulated following discrete uniform distribution. However, the stochastic CRB

for such a model turns out to be involved and approximate expressions were derived recently in [55],[54] for the cases of *only one and two sources* (i.e., $D = 1, 2$). Currently there exists no literature addressing the derivation of CRB for $D \geq N$ and it might as well be a topic of research in its own right.

4.9 Simulations

We now perform numerical study of the performance of the proposed $1D$ DOA estimation algorithm applied on a suitable nested array and compare it with the existing $2q$ MUSIC algorithm for $q = 2$ (i.e., using 4th order cumulants). We consider a total of $N = 6$ physical sensors and arrange them in form of a 4 level nested array with $N_1 = N_2 = N_3 = 2, N_4 = 3$, on which we apply the proposed algorithm. For applying $2q$ MUSIC, we arrange the 6 sensors in the form of a ULA and arrange the cumulants according to the orientation corresponding to $l = 1$. The source amplitudes are assumed to be statistically independent uniformly distributed (in particular, non Gaussian) BPSK symbols (i.e., each source amplitude takes up values ± 1 with probability $1/2$ each). The following points are to be noted:

- The 4 MUSIC algorithm cannot exploit the virtual elements of the 4th order difference set, they exploit a lower order virtual array, depending on the orientation. If we use the 4 level nested array for 4 MUSIC, the virtual array "seen" by the 4 MUSIC algorithm will *not* be an ULA and hence it will result in spurious peaks in the MUSIC spectrum. Since the virtual array of the ULA corresponding to $l = 1$ is still an ULA, the 4 MUSIC will not result in ambiguity in the estimated angles.

- The identifiability of the 4 MUSIC for the 6 sensor ULA and orientation $l = 1$ is $2.(6 - 1) = 10$ since the corresponding virtual array of the ULA has $2.(6 - 1) + 1 = 11$ elements. The identifiability of the proposed method on the 6 sensor 4 level nested array is $N_1N_2N_3N_4 - 1 = 23$. So, for fair comparison between the two methods, we will test both of them on a number of sources $D \leq 10$, so that they are identifiable by both.

4.9.1 Representative MUSIC Spectra

We first consider $D = 8$ sources with angles given by

$$\{-30^\circ, 60^\circ, 36^\circ, -60^\circ, 0^\circ, 45^\circ, -45^\circ, 15^\circ\} \tag{4.22}$$

and plot the representative MUSIC spectra for $SNR = 6$ dB. Fig. 4.3 (a) shows the MUSIC spectrum for the proposed method and (b) shows the MUSIC spectrum for 4 MUSIC for a snapshot value of $T = 800$. It can be clearly seen that for this value of snapshots, the 4 MUSIC cannot resolve all 8 sources whereas the proposed method can resolve all 8 sources. On increasing the number of snapshots to $T = 3000$, both the proposed method and 4 MUSIC are able to identify all 8 sources, as shown in Fig. 4.3 (c) and (d).

Both 4 MUSIC and the proposed method use subspace based DOA estimation techniques, which are superresolution by nature, i.e., they are not limited by the Rayleigh resolution limit of beamforming based approaches. We now plot representative MUSIC spectra to demonstrate the resolution performance of both the methods. We consider $D = 2$ sources with angles $\{30^\circ, 32^\circ\}$ separated by 2°, which is beyond the Rayleigh limit of the virtual arrays seen by both the methods. We consider $SNR = 6$ dB. Considering a snapshot value of $T = 2000$, the MUSIC spectra reveal that the proposed method can resolve these two sources whereas 4 MUSIC fails to do so, as shown in Fig. 4.4. However, at a higher snapshot value of $T = 5000$ snapshots, both the methods are able to resolve the two sources as shown in Fig. 4.4 (c) and (d). Note the sharpness of the peaks obtained from the proposed method, as compared to 4 MUSIC.

4.9.2 RMSE vs Snapshots

We now study the RMSE performance of the two methods for two different sets of source angles. For the first set, we consider $D = 6$ sources with angles equal to the first 6 values in (4.22) and for the second set, we consider $D = 8$ sources with angles given by (4.22). We consider SNR$= 6$ dB. We compute the RMSE of the D sources by averaging over 1000 Monte Carlo runs for each snapshot value. We finally plot the averaged RMSE vs snapshots, averaged over all D sources. The corresponding plots for $D = 6$ and $D = 8$ are given by Fig. 4.5 (a) and (b) respectively. Representative numbers for the averaged RMSE for selected values of snapshots are tabulated in Table 4.3 for $D = 6$. It can be clearly seen that the proposed method outperforms 4 MUSIC and the

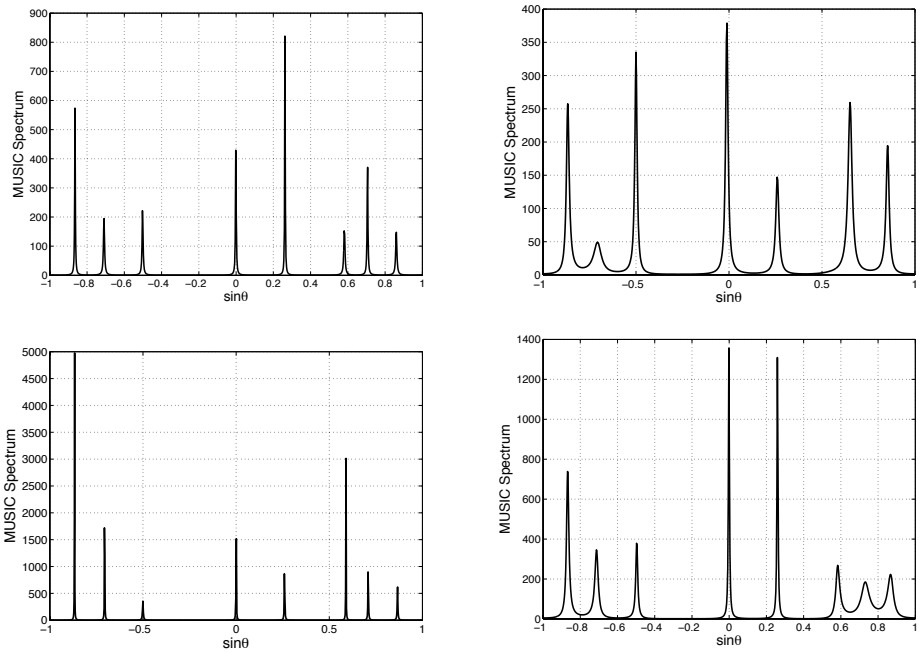


Figure 4.3: MUSIC Spectra for $D = 8$ sources, produced by (top left),(bottom left) Proposed Method on a 4 level nested array with 6 sensors and (top right),(bottom right) $2q$ MUSIC on a ULA with 6 sensors. Here $T = 800$, for (top left) and (top right), and $T = 3000$ for (bottom left) and (bottom right). SNR$= 6$ dB.

difference between the two methods is greater for the larger number of sources (i.e., $D = 8$). The performance of the two methods get closer for higher values of snapshots.

4.9.3 RMSE vs SNR

Next, we study the RMSE vs SNR performance of the two methods for the above two sets of source angles. We consider a snapshot value of $T = 2000$. The corresponding plots for $D = 6$ and $D = 8$ are given by Fig. 4.6 (a) and (b) respectively. Representative numbers for the averaged RMSE for selected values of SNR are tabulated in Table 4.4 for the case $D = 8$. In this case as well, the proposed method has lower RMSE compared to 4 MUSIC and the difference is larger for $D = 8$.

4.9.4 Model Perturbation

We now study the effect of model perturbation on the proposed method (in conjunction with 4 level nested array) and $2q$ MUSIC. We consider the first set of $D = 6$ DOAs. Denoting $\mathbf{a}(\theta_i)$ as the steering vector for ith source, we perturb it by adding a modeling error vector $\mathbf{e}_i$ to get the

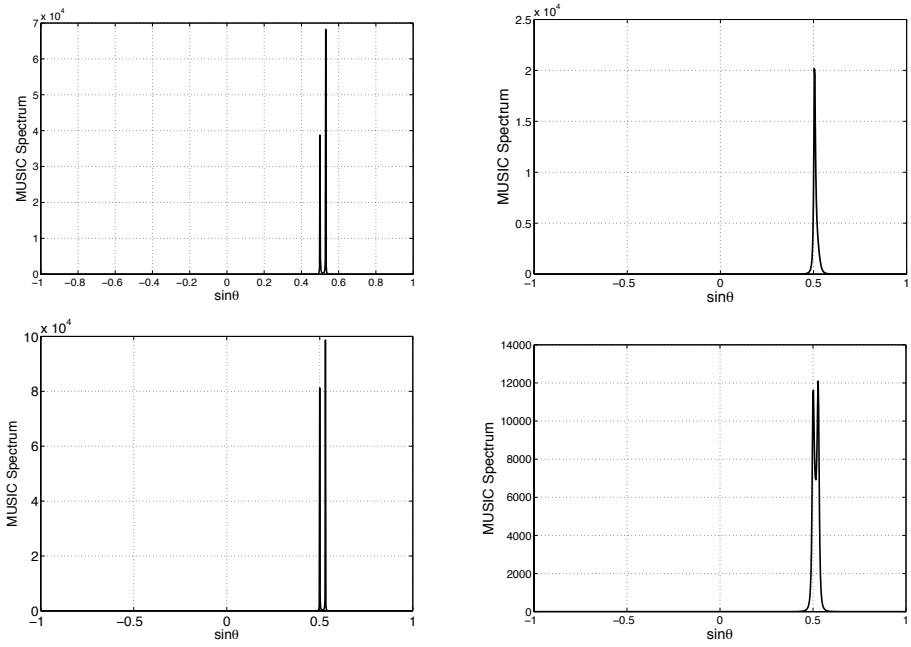


Figure 4.4: MUSIC Spectra for $D = 2$ sources, produced by (top left),(bottom left) Proposed Method on a 4 level nested array with 6 sensors and (top right),(bottom right) $2q$ MUSIC on a ULA with 6 sensors. Here $T = 2000$, for (top left) and (top right), and $T = 5000$ for (bottom left) and (bottom right). SNR$= 6$ dB.

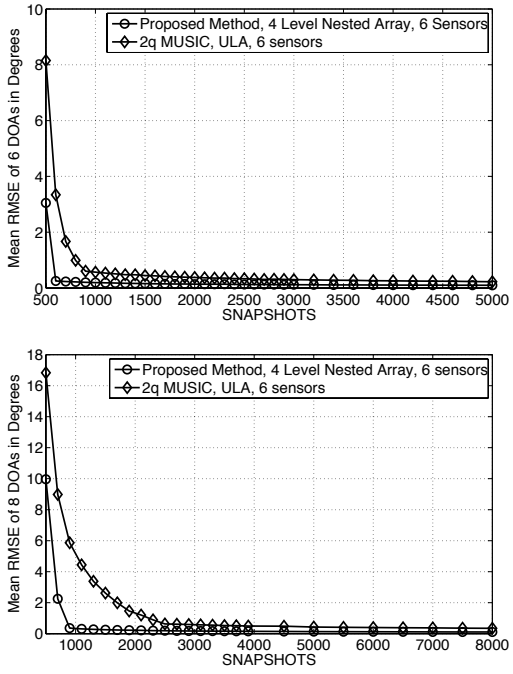


Figure 4.5: RMSE vs number of snapshots (T) of the proposed method and 4 MUSIC corresponding to (top) $D = 6$ sources, and (bottom) $D = 8$ sources. Here $N = 6$, SNR$= 6$ dB. No Model Perturbation.

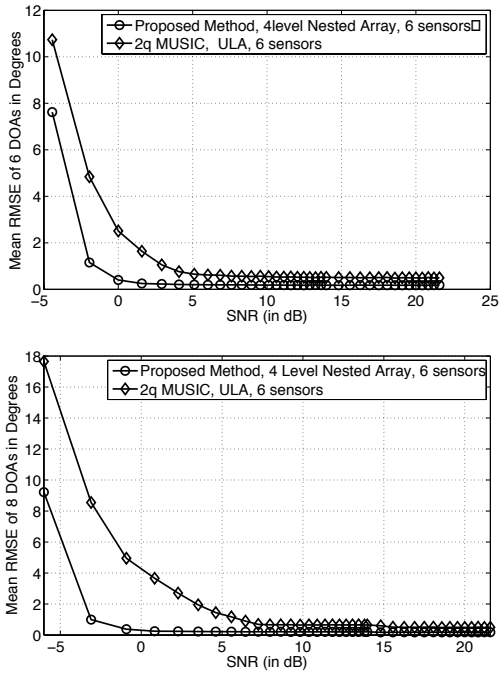


Figure 4.6: RMSE vs SNR (in dB) of the proposed method and 4 MUSIC corresponding to (top) $D = 6$ sources, and (bottom) $D = 8$ sources. Here $N = 6$, $T = 2000$. No Model Perturbation.

perturbed vector as

$$\tilde{\mathbf{a}}(\theta_i) = \mathbf{a}(\theta_i) + \mathbf{e}_i, \quad 1 \leq i \leq D$$

where $\mathbf{e}_i$ are assumed to be zero mean statistically independent Gaussian vectors with $E(\mathbf{e}_i \mathbf{e}_i^H) = \sigma_e^2 \mathbf{I}$ where $\sigma_e^2 = 0.0025$. Assuming SNR$= 6$ dB, the averaged RMSE obtained from the proposed method and the 4 MUSIC are plotted as a function of snapshots in Fig. 4.7. Representative numbers for the RMSE for selected snapshot values are also tabulated in Table 4.3. It can be seen that the proposed algorithm is more robust to modeling errors. In fact, the gap between the RMSE values of the two methods is found to be larger for the perturbed model as compared to the unperturbed case.

4.9.5 Probability of Resolution

As exhibited through the representative MUSIC spectra plots in Fig. 4.4, the proposed MUSIC appears to have better resolution power than 4 MUSIC. We consider the same two sources at 30° and 32° and estimate the probability of resolution as a function of the number of snapshots for both

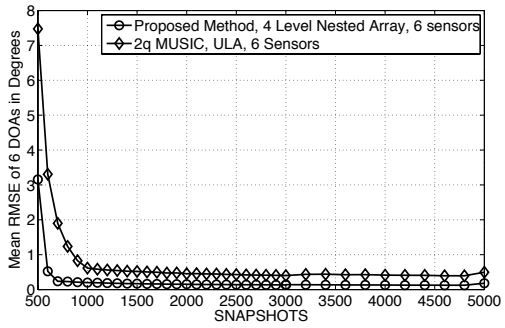


Figure 4.7: RMSE vs number of snapshots (T) of the proposed method and 4 MUSIC for the perturbed model. Here $N = 6$, SNR$= 6$ dB, $D = 6$.

the methods. Assuming SNR$= 6$ dB, we perform 1000 Monte Carlo runs for each snapshot value and count the number of times each method can resolve two sources. By dividing this number by 1000, we get an estimate of the probability of resolution of the two methods. The corresponding plot is given by Fig. 4.8 and representative numbers are tabulated in Table 4.5. The proposed method can resolve the two sources with a perfect resolution probability of 1 for all snapshot values. The 4 MUSIC performs poorly for low snapshot values and it reaches a probability of resolution of 0.9 at a very high snapshot value of $T = 7000$.

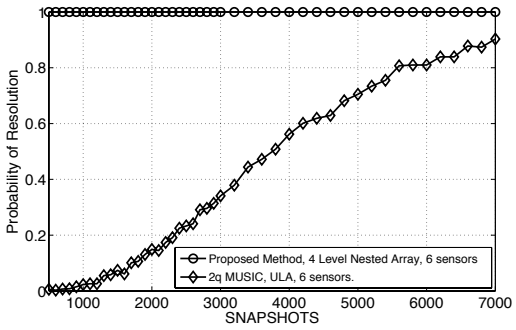


Figure 4.8: Probability of resolving the two sources at 30° and 32° of the proposed method applied on 4 level nested array and 4 MUSIC on ULA. Here $N = 6$, SNR$= 6$ dB.

4.10 Conclusion

In this chapter, we re-examined the $2q$th order cumulant based DOA estimation techniques and demonstrated how the identifiability of the earlier methods can be increased by looking at the $2q$th order difference co-array hidden in the cumulant matrices. We also proposed a new array

structure whose $2q$th order difference co-array provides $O(N^{2q})$ degrees of freedom when used in conjunction with $2q$th order statistics. An efficient algorithm for successfully estimating DOAs of $O(N^{2q})$ sources using N physical sensors was also proposed. The combination of the $2q$ level nested array and the proposed algorithm can identify $O(N^{2q})$ statistically independent sources which is much larger compared to the identifiability of the existing $2q$ MUSIC algorithm. In fact, the proposed algorithm was proved to attain the maximum identifiability obtained from $2q$th order cumulant based processing. The performance of the proposed algorithm and array geometry was numerically studied and compared with $2q$ MUSIC and the proposed method shows better RMSE performance for a given number of snapshots and SNR. It is also more robust to modeling errors and has a better resolution capacity. However, the proposed method does not apply to statistically dependent sources which can be handled by the $2q$ MUSIC method. Also, the proposed algorithm needs more computations than $2q$ MUSIC. So although the proposed method shows superior performance and identifiability for statistically independent sources, the $2q$ MUSIC might be a preferred choice in situations where the sources are statistically dependent and/or where computation cost is high. Future work in this area will be directed towards analyzing the performance of the proposed array structure and algorithm in terms of RMSE of DOA estimates, resolution, and robustness to modeling errors, which we have studied only numerically in this chapter.

4.11 Appendix

4.11.1 Virtual Sensors in the $2q$th order Difference set

4.11.1.1 Proof of Theorem 4.5.1

We will prove the result by induction on the order of the difference set. Let us denote the set of sensor positions (normalized with respect to $\lambda/2$) in the $2q$ level nested array as S_{2q}. The second order difference set of the $2q$ level virtual array is described by

$$\{u_1 - u_2\}, \quad u_1, u_2 \in S_{2q}. \tag{4.23}$$

Letting u_1 and u_2 take up values from the set $S_2 \subset S_{2q}$ where

$$S_2 = \{1, 2, \cdots, N_1 - 1\} \bigcup \{N_1 n, \quad 1 \leq n \leq N_2\},$$

it can be readily seen that $\{u_1 - u_2\}$ behaves like the difference co-array of a 2 level nested array with $N_1 - 1$ sensors in the first level and spacing N_1 in the second level. Hence, $\{u_1 - u_2\}$ contains all integers $[-N_1N_2 + 1, \cdots, 0, \cdots, N_1N_2 - 1]$ [135]. This forms the basis for the induction.

For a given integer k, let us define the set $S_k^{(ca)}$ as

$$S_k^{(ca)} = \{m, \quad -\prod_{i=1}^{k} N_i + 1 \leq m \leq \prod_{i=1}^{k} N_i - 1\}$$

and the set U_k as

$$U_k = \{\prod_{i=1}^{k-1} N_i n, \quad 1 \leq n \leq N_k\}.$$

Now assume that the $(2m-2)$th order difference set ($m \leq q$) of the $2q$ level nested array contains all integers given by the set $S_{2m-2}^{(ca)}$. We will show that this implies that the $2m$th order difference co-array contains all integers in the set $S_{2m}^{(ca)}$. To see this, notice the $2m$th order difference set of the $2q$ level nested array is given by the set $S_{2m}^{(ca)}$.

$$\{v_1 - v_2 + u_{2m-1}, \quad v_1, v_2 \in S_{2q}\} \tag{4.24}$$

where

$$u_{2m-1} \triangleq \sum_{i=1}^{m-1} u_i - \sum_{i=m}^{2m-2} u_i, \quad u_i \in S_{2q}$$

Then, under the induction assumption, u_{2m-1} takes up all integer values in the range given by the set $S_{2m-2}^{(ca)}$. Now, let v_2 take up values from U_{2m-1}. Then the set $\{v_2 - u_{2m-1}, v_2 \in U_{2m-1}, u_{2m-1} \in S_{2m-2}^{(ca)}\}$ contains the $2nd$ order difference set of a 2 level nested array with $\prod_{i=1}^{2m-2} N_i$ sensors (denoted by u_{2m-1}) in the first level and the second level has N_{2m-1} sensors (denoted by v_2) with intersensor spacing $\prod_{i=1}^{2m-2} N_i$. Hence it contains all integers in the set $S_{2m-1}^{(ca)}$. In a similar fashion, denoting $u_{2m} \triangleq v_2 - u_{2m-1}$, the $2m$th order difference set is rewritten as $\{v_1 - u_{2m}\}$. Letting v_1 take up values in the range from the set U_{2m}, it can be again seen that the set

$$\{v_1 - u_{2m}, v_1 \in U_{2m}, u_{2m} \in S_{2m-1}^{(ca)}\}$$

contains the $2nd$ order difference set of a 2 level nested array with $\prod_{i=1}^{2m-1} N_i$ sensors in the first

level and the second level has N_{2m} sensors (and intersensor spacing $\prod_{i=1}^{2m-1} N_i$). Hence it contains all integers in the set $S_{2m}^{(ca)}$.

4.11.1.2 Virtual sensors in the extra ULA segment

For each $1 < m \leq q$ in the above induction, consider again the set of differences $\{v_1 - u_{2m}\}$ where

$$v_1 = \prod_{i=1}^{2m} N_i, \quad u_{2m} \in S_{2m-1}^{(ca)}.$$

When u_{2m} takes up non positive values in $S_{2m-1}^{(ca)}$, i.e., values from the set

$$\{-(\prod_{i=1}^{2m-1} N_i - 1), \cdots, 0\},$$

the set of differences $\{v_1 - u_{2m}\}$ takes up all values

$$\{\prod_{i=1}^{2m} N_i + n, \quad 0 \leq n \leq \prod_{i=1}^{2m-1} N_i - 1\}$$

which adds an ULA segment on the positive end of the main ULA segment for each m. Due to symmetry of the $2q$ order difference set about 0, the corresponding negative values are also present, thereby adding the two extra ULA segments. Since this holds for each $1 < m \leq q$, we get the desired result by considering $m = q$.

Proof of Theorem 4.6.1

Given a feasible set of values $\{N_i\}$, let us consider any pair of integers N_k and N_l ($1 \leq k, l \leq 2q$) such that

$$N_k - N_l \geq 2.$$

Then, we can create a new pair of integers $N_k^{'}$ and $N_l^{'}$ so that

$$N_k^{'} = N_k - 1, \quad N_l^{'} = N_l + 1,$$

such that

$$N_k^{'}N_l^{'} = N_kN_l + (N_k - N_l - 1) > N_kN_l$$

Hence replacing N_k and N_l by $N_k^{'}$ and $N_l^{'}$ respectively strictly increases the objective function, while still satisfying the constraint (since $N_k^{'} + N_l^{'} = N_k + N_l$), as long as $N_k - N_l \geq 2$. Therefore, in the optimal solution $\{N_i^{opt}\}$, no two integers N_k^{opt} and N_l^{opt} can differ by more than 1. Since the solution to (4.17) is permutation symmetric, we can say that $\mathrm{N}_i^{opt} = m_1 + 1, \quad i = 1, 2, \cdots, n_1,$ $N_i^{opt} = m_1, \quad i = n_1 + 1, n_1 + 2, \cdots, 2q$ where m_1 and n_1 ($0 \leq n_1 \leq 2q - 1$) are integers satisfying $\sum_{i=1}^{2q} N_i^{opt} = N + 2q - 1$, implying

$$2qm_1 + n_1 = N + 2q - 1 = 2qm + n$$

Since $0 \leq n, n_1 \leq 2q - 1$, it follows from the uniqueness of remainder of $N + 2q - 1$ modulo $2q$, that

$$n = n_1 \quad and \quad m = m_1$$

Substituting the optimal values in the objective function, it can be easily verified that the maximized value is $2m^{2q-n}(m + 1)^n - 1$.

Table 4.1: **Summary of Proposed Spatial Smoothing Based Algorithm.**

Given	Received Signal Sequence $\{\mathbf{y}(t)\}_{t=1}^{T}$ using a $2q$ level nested array, number of sources $= D$.
Step 1	Compute estimates of $2q$th order cumulant of the received data $\mathbf{y}(t)$ using (4.5) and replacing the expectations by corresponding temporal snapshot averages (averaged over T snapshots). A total of N^{2q} cumulant values are computed which are arranged in the form of the vector $\mathbf{c}_{\text{vec}} \in \mathcal{C}^{N^{2q}}$.
Step 2	Compute the $2q$th order difference set of the physical array. Let $2M_{2q}+1$ denote the number of elements in its ULA segment (always odd), extending from $-M_{2q}$ to M_{2q}. Extract the rows of $\mathbf{c}_{\text{vec}}$ which corresponds to the elements in the ULA segment of the $2q$th order difference co-array, to obtain the vector $\mathbf{z}_1 \in \mathcal{C}^{2M_{2q}+1}$. Order the elements in $\mathbf{z}_1$ such that the ith row corresponds to the element in the ULA segment at location $-M_{2q}+i-1$.
Step 3	Divide the vector $\mathbf{z}_1$ into $M_{2q}+1$ overlapping subvectors $\mathbf{z}_{1i} \in \mathcal{C}^{(M_{2q}+1)\times 1}, i = 1, 2, \cdots, M_{2q}+1$ such that $[\mathbf{z}_{1i}]_m = [\mathbf{z}_1]_{i+m-1}, \quad 1 \le m \le M_{2q}+1$
Step 4	Construct the positive semidefinite matrix $\mathbf{C}_{virtual} \in \mathcal{C}^{(M_{2q}+1)\times(M_{2q}+1)}$ as $\mathbf{C}_{virtual} = \frac{1}{M_{2q}+1} \sum_{i=1}^{M_{2q}+1} \mathbf{z}_{1i}\mathbf{z}_{1i}^{H}.$
Step 5	Apply subspace based algorithms like MUSIC on $\mathbf{C}_{virtual}$ in the same way as applied on the covariance matrix of a $M_{2q}+1$ element ULA, to perform DOA estimation of upto $D \le M_{2q}$ sources.

htb

Table 4.2: **Comparison of Proposed Algorithm and $2q$ MUSIC for $1D$ DOA Estimation**

	Virtual array Exploited	**Required Geometry**	**Maximum Identifiability**	**Handles Correlated sources**	**Complexity**
Proposed	$2q$th order difference set	$2q$th order difference set needs to be ULA	$O(N^{2q})$	No	$O(N^{6q})$ computations
$2q$ **MUSIC**	Different virtual arrays, characterized by orientation l. Can exploit upto qth order difference set for even q.	Virtual array can be ULA or non ULA. Ambiguity arises for non ULA.	$O(N^q)$	Yes	$O(N^{3q})$ computations.

Table 4.3: **RMSE (in degrees) vs Snapshots T, $D = 6$ Sources**

		$T = 600$	$T = 800$	$T = 1000$	$T = 4000$
No Perturbation	**RMSE (Proposed)**	0.2527	0.2185	0.1931	0.1044
	RMSE ($2q$ MUSIC)	3.34	0.99	0.5686	0.2575
With Model Perturbation	**RMSE (Proposed)**	0.5227	0.2289	0.1996	0.1308
	RMSE ($2q$ MUSIC)	3.3	1.237	0.6172	0.4144

Table 4.4: **RMSE (in degrees) vs SNR, $D = 8$ Sources, No Perturbation**

	SNR= 0.83 **dB**	**SNR=** 4.6 **dB**	**SNR=** 7.2 **dB**	**SNR=** 15.5 **dB**
RMSE (Proposed)	0.249	0.22	0.209	0.177
RMSE ($2q$ MUSIC)	3.66	1.44	0.686	0.508

Table 4.5: **Probability of Resolution vs Snapshots, $D = 2$ Sources, No Perturbation**

	$T = 1000$	$T = 3000$	$T = 5000$	$T = 7000$
Probability (Proposed)	1	1	1	1
Probability ($2q$ MUSIC)	0.02	0.34	0.705	0.9

Chapter 5

Wideband Array Processing

5.1 Introduction

In this chapter, we will consider a wideband model for the signal received by an antenna array, as opposed to the narrowband model that has been so far studied in Chapters 2, 3 and 4. We will develop new algorithms for two problems in wideband array processing, namely, DOA estimation of wideband signals and frequency invariant beamforming.

The first part of this chapter considers the problem of estimating the directions-of-arrival (DOA) of signals impinging on a sensor array. This problem has widespread applications in source localization for radar, sonar and wireless communication systems. When the incoming signals are wideband, the most popular approach is to decompose the wideband signal into narrowband components and wideband DOA estimation in that case is mainly concerned with finding a clever way to use the multiple correlation matrices at different frequencies to get accurate DOA estimates. The incoherent signal subspace method (ISSM) [196]is the simplest wideband method which estimates the source DOAs separately at each narrowband frequency and then constructs the final estimate by taking an average. While this works well at high SNR, the performance can suffer severely at low SNR because even a single outlier from one narrowband component can potentially lead to inaccurate estimates through the averaging process. Also it is incapable of handling coherent sources. To overcome these drawbacks, several coherent methods have been suggested [191, 64, 106, 46, 93]. Of these, the coherent signal subspace method (CSSM) [191] was the earliest which aimed at focusing the signal subspace at each frequency to that at a certain reference focusing frequency and then construct a single correlation matrix using these focused narrowband correlation matrices on which narrowband DOA estimation algorithms can be applied. This method has good performance at low

SNR and can also handle coherent sources [191]. However, construction of the focusing matrices needs preliminary estimates of DOAs and the performance of CSSM is sensitive to the error in the preliminary estimates, which can give rise to biased final estimates [171]. The weighted average of signal subspaces (WAVES) [46] is another coherent method and it also requires focusing matrices. Though it can avoid initial estimates by using BI-CSSM technique, its performance is worse than when it uses focusing matrices with good initial estimates. Recently, a novel wideband DOA algorithm called test of orthogonality of projected subspaces (TOPS) [204] has been proposed. It is in between the coherent and incoherent processing, performing best at low to mid SNR and does not require initial estimates. However, the spatial spectrum used in TOPS can show false peaks which is a disadvantage of this method. It is seen that coherent methods are definitely advantageous in terms of good performance at low SNR and capability for handling coherent sources. However, the dependence of the focusing matrices used in coherent methods, on the preliminary DOA estimates is a severe drawback of these methods which gives rise to increased computation and also potentially biased estimates. To overcome this problem, in this chapter, we suggest a new technique to focus the signal subspaces at different frequencies, which does not require the preliminary DOA estimates. This technique, which we call autofocusing, constructs the focusing matrices entirely on the basis of the received data and does not require knowledge of array manifold. Also, the focusing matrices are made unitary by construction so that there is no focusing loss.

The second part of the chapter focuses on beamforming for wideband signals. In array processing for wideband signals, the temporal frequency and the spatial angle of the impinging signals are usually coupled together in the phase, which makes the resulting beampattern a function of both frequency and spatial angle. To realize a desired beampattern in such cases, tapped-delay line filters are needed following each sensor. The number of taps needed to generate a beampattern, increases with the bandwidth of the signal, the number being one per sensor for the case of narrowband beamforming. This increase in the processing overhead for wideband signals can potentially lead to increased convergence time and poor numerical properties of the adaptive algorithms, when these tap weights need to be adapted in real time. Hence, as an alternative approach, frequency invariant beamforming is often used to make the resulting beampattern approximately independent with respect to the temporal frequency, thereby reducing all further processing to that of usual narrowband array processing. It has been studied extensively because of the advantages it offers in terms of reduced complexity for adaptive beamforming, broadband DOA estimation and

so forth [31, 115, 161, 112, 192, 193, 194]. Several techniques exist in literature discussing generation of these frequency invariant beams [31, 115, 161, 193]. However, all of them either use filters or frequency dependent weights following each sensor.

In [82], however, it has been shown that frequency invariant beamforming can be achieved using just one weight per sensor instead of tapped delay-lines, by extending the dimension of the array from linear to rectangular. We review this idea briefly in Section 5.6. In this chapter, we generalize this concept to the case of more generic 2 dimensional arrays with elements on a possibly non separable lattice (as opposed to the rectangular array which is an example of a separable lattice) and demonstrate how to perform frequency invariant beamforming by using the extra spatial dimension provided by these lattices, to get rid of temporal processing. In particular, we study the effect of different lattice geometries (encompassing the special cases of a ULA and a rectangular array) aiming to gain insight into how they affect the approximation of the frequency invariant beams. We describe how, using only one constant weight at the output of each sensor, frequency invariant beamforming can be achieved by designing the beam in a (non linear) transform domain (instead of the original angle-frequency domain) where a simple DFT relation exists between the weights to be designed and the beampattern to be approximated. However, the lattice geometry plays an important role in deciding what the domain of design looks like in the transformed domain. To exploit this effect to the maximum extent, we propose a new approach to sampling the beampattern in the transform domain and the simulations based on this approach reveal that non separable lattices might be a better choice in terms of approximating the beampattern, the reason for which is also qualitatively explained. As an application of the frequency invariant beamforming, we demonstrate how to perform wideband DOA estimation using these frequency invariant beams, using only narrowband subspace based approaches.

The performance of the frequency invariant beamformers however depends on the size of the array and for a 2 dimensional array, it could mean the use of a large number of sensors. A novel solution to this is proposed by using the so called difference co-array of a physical array. The concept of difference co-array has been explored previously in detail in [89] and the references therein. We propose a new approach to beamforming based on this difference co-array which essentially performs spatial filtering of the power of the signal (instead of amplitude). This non linear approach to beamforming, as we shall show, enables us to exploit the additional degrees of freedom provided by the co-array of the actual array and we shall precisely use this to avoid using

additional sensors. We first prove that any two dimensional array of size $M \times N$ *with elements on a lattice*, can be generated using 2 suitable ULAs with $M + N$ elements. We then propose a novel idea of producing the same frequency invariant beampattern as generated by an actual lattice, using the proposed beamforming approach based on the difference co-array. The additional freedom provided by the extra spatial dimension is used to get rid of TDL filters in wideband frequency invariant beamforming.

Portions of this chapter have been presented in [132] and [131].

5.2 Outline

The rest of the chapter is organized as follows. In Section 5.3, we briefly review some important wideband DOA estimation algorithms. In Section 5.4, we discuss the proposed autofocussing method. In Section 5.5, we provide numerical examples to compare the proposed method with existing wideband direction finding algorithms. In Section 5.6, the wideband beamforming model is introduced and the problem statement is explained. In Section 5.7, we introduce the theory of frequency invariant beamforming using 2 dimensional arrays with elements on a lattice. We provide design examples and study the effect of lattice geometry on the frequency invariant beamforming. In Section 5.8, we briefly review how to generate virtual arrays with elements on a lattice using difference co-array of $1D$ arrays. We then propose the design of frequency invariant beams using the difference co-array of suitable combination of ULAs, to gain the same $O(N^2)$ degrees of freedom of a lattice using only $O(N)$ physical sensors.

5.3 Background of Wideband Beamforming

5.3.1 Signal Model

Consider an N element linear array which has the property that $D < N$ steering vectors of the array corresponding to D different source angles are linearly independent, spanning a D dimensional subspace (also known as the signal subspace). Consider $D < N$ wideband sources $s_1(t), s_2(t), \cdots, s_D(t)$ impinging on the array from directions of arrival $\theta_1, \theta_2, \cdots, \theta_D$ respectively. The signal received at

the nth sensor is given by

$$x_n(t) = \sum_{d=1}^{D} s_d(t - \tau_n(\theta_d)) + \eta_n(t), n = 1, 2, \cdots, N \tag{5.1}$$

where $\tau_n(\theta_d)$ is the propagation delay associated with the dth source and the nth sensor and $\eta_n(.)$ is the additive noise at the nth sensor. The incoming signal at each sensor is first sampled at frequency f_s and the samples are then partitioned into segments of $K = \Delta T f_s$ samples each. Then, a K-point DFT is applied to the K samples in a segment. The DFT coefficients from the N sensors can be expressed as

$$\mathbf{X}[j] = \mathbf{A}(f_j)\mathbf{S}[j] + \mathbf{N}[j], \quad j = 0, 1, \cdots, K-1 \tag{5.2}$$

where

$$f_j = \frac{j}{K} f_s, \tag{5.3}$$

and $\mathbf{X}[j] = \begin{bmatrix} X_1[j] & X_2[j] & \cdots & X_N[j] \end{bmatrix}^T$ with $X_n[j]$ denoting the jth DFT coefficient of samples of $x_n(t)$. Also,

$$\mathbf{A}(f_j) = \begin{bmatrix} \mathbf{a}_{\theta_1}(f_j) & \mathbf{a}_{\theta_2}(f_j) & \cdots & \mathbf{a}_{\theta_D}(f_j) \end{bmatrix}. \tag{5.4}$$

with

$$\mathbf{a}_{\theta_d}(f_j) = \left[e^{-j2\pi f_j \tau_1(\theta_d)}, \cdots, e^{-j2\pi f_j \tau_N(\theta_d)} \right]^T \tag{5.5}$$

as the steering vector of the array at frequency f_j for the dth source.

$$\mathbf{N}[j] = \begin{bmatrix} N_1[j] & N_2[j] & \cdots & N_N[j] \end{bmatrix}^T. \tag{5.6}$$

$$\mathbf{S}[j] = \begin{bmatrix} S_1[j] & S_2[j] & \cdots & S_D[j] \end{bmatrix}^T \tag{5.7}$$

where $N_n[j]$ and $S_d[j]$ are the jth DFT coefficients of $\eta_n(t)$ and $s_d(t)$ respectively. Under the assumption that ΔT is long enough compared to the correlation time of the signals and the noise so that we can regard the DFT coefficients as uncorrelated [64, 191], we can write

$$\mathbf{R}_{xx}[j] \triangleq E[\mathbf{X}[j]\mathbf{X}^H[j]] = \mathbf{A}(f_j)\mathbf{R}_{ss}[j]\mathbf{A}(f_j)^H + \sigma_n^2 I. \tag{5.8}$$

Here $\mathbf{R}_{ss}[j] = E[\mathbf{S}[j]\mathbf{S}^H[j]]$ and we have assumed the noise samples to be zero mean temporally and spatially white Gaussian random process with variance σ_n^2 so that the autocorrelation matrix at each DFT frequency is $\sigma_n^2 I$. Assuming that the D sources are uncorrelated, the $D \times D$ matrix $\mathbf{R}_{ss}[j]$ has full rank. Then, if the eigenvectors of $\mathbf{R}_{xx}[j]$ are ordered in decreasing order with respect to their eigenvalues, the first D eigenvectors span the same subspace of dimension D (known as signal subspace at frequency f_j) as $\mathcal{R}(\mathbf{A}(f_j))$ and are known as the signal subspace eigenvectors. The last $N - D$ eigenvectors are called the noise subspace eigenvectors and they span a subspace of dimension $N - D$ orthogonal to the signal subspace.

5.3.2 Review of Coherent and Incoherent Methods of Wideband DOA Estimation

Incoherent signal subspace method (ISSM) [196] is the simplest wideband DOA estimation method which estimates the signal and noise subspace at each frequency independently and averages the estimated DOAs from each frequency bin to get the final estimate. This method, as mentioned in the earlier section, suffers at low SNR [191]. The coherent method [191] attempts at solving this problem by combining the signal subspaces *coherently* into one single autocorrelation matrix to which narrowband high resolution methods (such as MUSIC) can be applied. The key idea is to design a focusing matrix $\mathbf{T}(f_j)$ which transforms the array manifold at frequency f_j to that at a common frequency f_0 as follows

$$\mathbf{T}(f_j)\mathbf{A}(f_j) = \mathbf{A}(f_0). \tag{5.9}$$

Then the coherent autocorrelation matrix is generated as

$$\mathbf{R}_{coh} = \sum_{j=0}^{K-1} \mathbf{T}(f_j)\mathbf{R}_{xx}[j]\mathbf{T}^H(f_j) = \mathbf{A}(f_0)\mathbf{R}_s\mathbf{A}^H(f_0) + \sigma_n^2 \sum_{j=0}^{K-1} \mathbf{T}(f_j)\mathbf{T}^H(f_j) \tag{5.10}$$

where $\mathbf{R}_s = \sum_{j=0}^{K-1} \mathbf{R}_{ss}[j]$. Now any standard narrowband DOA estimation method such as MUSIC [160], can be applied by computing the eigenvectors of the matrix pencil $(\mathbf{R}_{coh}, \sum_{j=0}^{K-1} \mathbf{T}(f_j)\mathbf{T}^H(f_j))$. The coherent method is potentially capable of better performance even at low SNR as it first focuses the energy at different frequencies at the focusing frequency and then performs narrowband DOA estimation at that focusing frequency. Also by the frequency averaging process, it can handle correlated sources. WAVES [46] is another interesting coherent method which, instead of using the entire autocorrelation matrix at different frequencies, uses sum of suitably weighted signal subspaces. However, it also requires focusing matrices.

The focusing matrices are non-unique and several discussions on construction of these focusing matrices can be found in [191, 64, 93]. The most popular way to generate these focusing matrices is to construct a unitary focusing matrix by minimizing the Frobenius norm of the manifold mismatches, viz.,

$$\min_{\mathbf{T}(f_i)} ||\mathbf{A}(f_0) - \mathbf{T}(f_j)\mathbf{A}(f_j)||_F \tag{5.11}$$

$$\text{subject to } \mathbf{T}(f_i)\mathbf{T}^H(f_i)) = I.$$

This class of focusing matrices is called rotational signal-subspace (RSS) focusing matrices [93] and the matrix is required to be unitary in order to preserve the SNR before and after focusing. The focusing matrices are further discussed in [64] where it is shown that the focusing matrices (including the RSS matrices) belong to the broader class of matrices called signal-subspace transformation (SST) matrices.

However, to construct the focusing matrices following any of the popular methods, it is crucial to have some preliminary estimate of the DOAs. Poor initial estimates might lead to inaccurate focusing and gives rise to final DOA estimates which are biased [171]. The need for initial DOA estimates can be avoided by the BI-CSSM method [106] which utilizes a set of frequency invariant beams spanning the field-of-view (FOV). However the performance of this method heavily depends on the size of the FOV and the array geometry.

5.4 Autofocusing approach to coherent signal subspace based wide-band DOA estimation

In this chapter, we propose a novel technique of generating focusing matrices, which *do not require any* preliminary DOA estimation, and yet can coherently combine the signal subspaces at different frequencies into one coherent subspace at the chosen frequency, thereby assuring all the advantages offered by the CSS method. The need for preliminary DOA estimates is avoided by noticing the simple fact that the signal subspace eigenvectors span the same subspace as the array manifold at each frequency and so, we can use them to form the focusing matrix, instead of using the array steering vectors, which require the preliminary knowledge of the DOAs.

Let us consider the same signal model as described in Section I. In particular, the autocorrelation matrix for the ith DFT coefficient is

$$R_{xx}[i] = A(f_i)R_{ss}[i]A(f_i)^H + \sigma_n^2 I. \tag{5.12}$$

We propose the following matrix for focusing the frequency f_i onto the frequency f_0:

$$T_{auto}(f_i) = \frac{1}{\sqrt{K}} U(f_0)U(f_i)^H \tag{5.13}$$

where K is the number of frequency bins considered and $U(f_i)$ is a $N \times N$ unitary matrix whose columns are the eigenvectors of the autocorrelation matrix $R_{xx}[i]$. The motivation behind this choice will be soon made clear. It is to be noted that this matrix belongs to the same class of SST matrices as proposed by Doron in [64]. Since the eigenvectors of $R_{xx}[i]$ can be grouped into signal space and noise space eigenvectors, we can represent $U(f_i)$ as

$$U(f_i) = \begin{bmatrix} U_{ss}(f_i) & U_N(f_i) \end{bmatrix} \tag{5.14}$$

where $U_{ss}(f_i)$ is a $N \times D$ matrix whose columns represent the D orthonormal eigenvectors of $R_{xx}[i]$ corresponding to the D largest eigenvalues, and $U_N(f_i)$ is a $N \times (N-D)$ matrix whose columns represent the remaining $(N-D)$ orthonormal eigenvectors of $R_{xx}[i]$. It is to be noted that to construct $T_{auto}(f_i)$, we do not need to know the number of sources, D, since we are using the complete set of eigenvectors of $R_{xx}[i]$ as columns of $U(f_i)$, as opposed to using, say, D of them.

However this does not prevent us from representing $U(f_i)$ as in (5.14) because we can always construct $U(f_i)$ by making the jth column equal to the eigenvector corresponding to the jth largest eigenvalue. We can now multiply $X[i]$ by the focusing matrix and define the transformed vector

$$Y[i] = T_{auto}(f_i)X[i]. \tag{5.15}$$

Taking the autocorrelation of $Y[i]$, we get,

$$\begin{aligned} R_{yy}[i] &= E[Y[i]Y[i]^H] \\ &= T_{auto}(f_i)R_{xx}[i]T^H_{auto}(f_i) \\ &= T_{auto}(f_i)A(f_j)R_{ss}[i]A^H(f_i)T^H_{auto}(f_i) + \sigma_n^2 T_{auto}(f_i)T^H_{auto}(f_i) \\ &= T_{auto}(f_i)A(f_i)R_{ss}[i]A^H(f_i)T^H_{auto}(f_i) + \sigma_n^2 I \end{aligned} \tag{5.16}$$

where (5.16) is due to the fact that $T_{auto}(f_i)$, being a product of two unitary matrices, is itself unitary, so the noise autocorrelation at frequency f_i remains unchanged. Now we have,

$$\begin{aligned} & T_{auto}(f_i)A(f_i) \\ &= \frac{1}{\sqrt{K}} \begin{bmatrix} U_{ss}(f_0) & U_N(f_0) \end{bmatrix} \begin{bmatrix} U^H_{ss}(f_i) \\ U^H_N(f_i) \end{bmatrix} A(f_i) \\ &= \frac{1}{\sqrt{K}} \begin{bmatrix} U_{ss}(f_0) & U_N(f_0) \end{bmatrix} \begin{bmatrix} U^H_{ss}(f_i)A(f_i) \\ 0 \end{bmatrix} \qquad (5.17) \\ &= \frac{1}{\sqrt{K}} U_{ss}(f_0)U^H_{ss}(f_i)A(f_i) \qquad (5.18) \end{aligned}$$

where the last line of (5.17) is due to the fact that the noise eigenvectors at frequency f_i are orthogonal to the array manifold $A(f_i)$ at frequency f_i. Substituting (5.18) into (5.16), we get,

$$R_{yy}[i] = \frac{1}{K}U_{ss}(f_0)\tilde{R}[i]U^H_{ss}(f_0) + \frac{1}{K}\sigma_n^2 I \tag{5.19}$$

where

$$\tilde{R}[i] = \frac{1}{K}U^H_{ss}(f_i)A(f_i)R_{ss}[i]A^H(f_i)U_{ss}(f_i). \tag{5.20}$$

We next sum (5.19) corresponding to the K frequencies to get the coherently combined autocorrelation matrix as

$$\begin{aligned} R_{coh} &= \sum_{i=0}^{K-1} R_{yy}[i] \\ &= \frac{1}{K} U_{ss}(f_0)(\sum_{i=0}^{K-1} \tilde{R}[i]) U_{ss}^H(f_0) + \sigma_n^2 I \\ &= U_{ss}(f_0) R_s U_{ss}^H(f_0) + \sigma_n^2 I \end{aligned} \tag{5.21}$$

where

$$R_s = \frac{1}{K} \sum_{i=0}^{K-1} \tilde{R}[i] \tag{5.22}$$

R_{coh} plays the role of the universal focused sample correlation matrix, where the signal subspaces at different frequencies are focused to the signal subspace at f_0 which is spanned by the columns of $U_{ss}(f_0)$. It can be seen that (5.21) has similar form as (5.10). However, instead of the actual array manifold $A(f_0)$ as the focusing array manifold, we have the matrix $U(f_0)$ whose columns span the same subspace as $A(f_0)$ and hence, as we shall show shortly, they carry the same information about the focusing signal subspace. This is the primary reason why, in our proposed method, the focusing can be performed without the preliminary DOA estimates. The signal autocorrelation matrices at different frequencies are, transformed and then summed to get the total correlation matrix R_s whose rank will be discussed in the following theorem. As the focusing matrices are unitary, the spatial whiteness of the noise is preserved and hence the noise autocorrelation matrix remains unchanged. Thus, (5.21) has the form of a narrowband sample autocorrelation matrix from which we can extract the DOA estimates following similar technique as narrowband MUSIC algorithm, as explained by the following theorem.

Theorem 5.4.1. *: Let R_{coh} be the coherently combined sample autocorrelation matrix of the sensor outputs as given by (5.21). Then the following statements are true:*

(i) R_s is a non negative definite matrix satisfying

$$\max_i \{rank(R_{ss}[i])\} \leq rank(R_s) \leq D. \tag{5.23}$$

If the sources are uncorrelated, then R_s is a positive definite matrix, with

$$rank(R_s) = D \tag{5.24}$$

(ii) Let e_i and λ_i, $i = 1, 2, \cdots, N$ be the eigenvectors and the corresponding eigenvalues, (in decreasing order), of the matrix R_{coh}. Then, under the assumption that the sources are uncorrelated, we have

(a) $\lambda_{D+1} = \lambda_{D+2} = \cdots = \lambda_N = \sigma_n^2$

(b) Column span of $E_N \triangleq \begin{bmatrix} e_{D+1} & e_{D+2} & \cdots & e_N \end{bmatrix}$ is orthogonal to Range space of $A(f_0)$.

□

Proof. (i) Recall,

$$\tilde{R}[i] = \frac{1}{K} U_{ss}^H(f_i) A(f_i) R_{ss}[i] A^H(f_i) U_{ss}(f_i) \tag{5.25}$$

Since $U_{ss}(f_i)$ and $A(f_i)$ are $N \times D$ matrices with full column rank, spanning the same range space, $U_{ss}^H(f_i)A(f_i)$ is a full-rank $D \times D$ matrix and so

$$rank(\tilde{R}[i]) = rank(R_{ss}[i]) \tag{5.26}$$

Moreover, as $R_{ss}[i]$ is non negative definite (allowing coherent sources), so must be $\tilde{R}[i]$. Therefore $R_s = \sum_{i=0}^{K-1} \tilde{R}[i]$, being a sum of non negative definite matrices, is itself non negative definite. Also, it is easy to check that

$$\mathcal{N}(R_s) = \bigcap_{i=0}^{K-1} \mathcal{N}(\tilde{R}[i]) \tag{5.27}$$

and hence, rank of the $D \times D$ matrix R_s satisfies

$$\max_i \{rank(R_{ss}[i])\} \leq rank(R_s) \leq D \tag{5.28}$$

When the sources are uncorrelated, then $R_{ss}[i]$ is positive definite for $i = 0, 1, \cdots, K-1$. So, using (5.26), we can say that $rank(\tilde{R}[i]) = D \ \forall i$. This, along with (5.28), implies that $rank(R_s) = D$ and hence R_s becomes a full rank positive definite matrix.

(ii) When the sources are uncorrelated, we know from (5.24), that R_s is a positive definite matrix. Since $U_{ss}(f_0)$ is $N \times D$ matrix with full column rank D, so $U_{ss}(f_0)R_sU_{ss}^H(f_0))$ is a non negative definite rank-D matrix, whose columns span the same subspace as $\mathcal{R}(A(f_0))$ since $\mathcal{R}(U_{ss}(f_0)) = \mathcal{R}(A(f_0))$. The diagonal noise autocorrelation matrix in (5.21) makes R_{coh} positive definite which is essentially the sum of a rank-P non negative definite matrix and a diagonal matrix. The eigenstructure of R_{coh} is exactly the same as that of the sample autocorrelation matrix encountered in narrowband MUSIC algorithm and hence the proof of this part of the theorem follows the same lines as given in [160].

□

The theorem ensures that we can perform narrowband MUSIC on the coherently combined covariance matrix obtained by using our proposed focusing matrix. We have explicitly proved that all D sources can be identified when they are uncorrelated. However, this condition can be relaxed to include even correlated sources. In this case, though the source covariance matrices $R_{ss}[i]$ are singular, the singularity may be removed by the frequency averaging process so that the averaged matrix R_s becomes non singular.

Advantages of the proposed autofocusing technique:

We now summarize the advantages of our proposed focusing method:

(i) It does not need any initial estimate of the DOAs to construct focusing matrix. Thus it completely avoids the need for preprocessing.

(ii) The focusing matrix $T_{auto}(f_j)$ is a unitary matrix by construction, and hence, according to [64, 93], it results in no focusing loss. Also, it preserves the spatial whiteness of the noise, so that the coherently combined noise autocorrelation matrix remains diagonal.

(iii) Construction of the focusing matrix at a particular frequency does not require the knowledge of the number of sources since we use the full set of eigenvectors (as opposed to using only the signal-subspace eigenvectors) of the autocorrelation matrix at that frequency. This in turn avoids the errors that might occur due to wrong detection of number of sources at low SNR locally at a given frequency.

(iv) The focusing matrices are constructed *entirely on the basis of the received data*, by the eigendecomposition of the signal autocorrelation matrix at different frequencies. So it can be viewed

as a fully adaptive method. Most importantly, since we do not use the array manifold explicitly in constructing the focusing matrices, it does not require accurate calibration and can avoid the errors that occur due to mismatches.

5.5 Simulation Examples

In this section, the performance of the proposed method is evaluated and compared against popular wideband DOA estimation algorithms through simulation examples. We consider $D = 3$ broadband sources impinging on a 10 element ULA from 0°, 30° and 45°. The sampling frequency is chosen to be twice the highest frequency and hence, the sources occupy the frequency band between $\pi/2$ and π in the digital frequency domain. The sensor spacing is chosen to be $\lambda_{min}/2$ where λ_{min} is the wavelength corresponding to the highest frequency in the signal. This spacing ensures that there is no aliasing in the spatial domain.

The samples of the impinging signal is divided into $M = 100$ segments of $K = 256$ samples each. In each segment the 256 samples are converted to frequency domain by a 256-point DFT which are then processed using four different algorithms: i) The proposed autofocusing method. ii) CSSM using RSS focusing matrices [191, 93] with perfect intial DOA estimates (named CSSM I) iii) CSSM using RSS focusing matrices with intial DOA estimates obatined by delay-and-sum beamforming (named CSSM II), and iv) the recently proposed TOPS [204] algorithm. The RSS focusing matrices are constructed following [191] and the focusing frequency for both the CSSM methods and the proposed method is chosen to be the middle frequency of the band.

The performances of the four algorithms are compared for the source at 45° in terms of their RMSE in Fig. 5.1 and bias in Fig. 5.2 v/s SNR. The proposed autofocusing technique is found to outperform CSSM II and TOPS with respect to both RMSE and bias. The CSSM method with perfect initial DOA estimates exhibits the best performance but it is impractical. It is to be noted that CSSM II, which is the more practical way to implement CSSM, suffers from bias even at high SNR which is due to inaccurate initial DOA estimates. The proposed method, however, not requiring any such initial estimates, gives rise to unbiased estimates as evident from the plots.

It is also to be noted that TOPS spectrum can show spurious peaks at angles other than the true DOAs. To exhibit this, the TOPS spectrum (inverse of the smallest singular value) and the MUSIC spectrum for the proposed method are plotted (as a function of angle of arrival) at SNR=0 dB. While

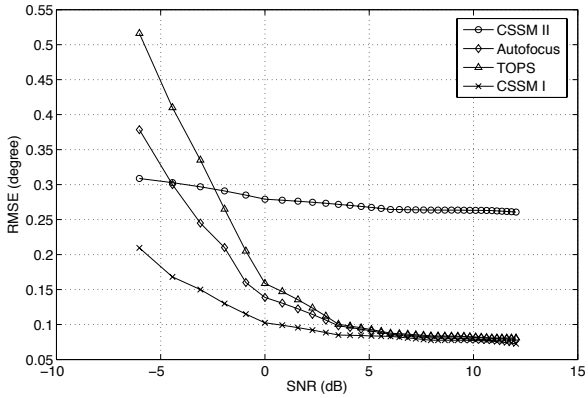


Figure 5.1: Comparison of RMSE of the different wideband DOA estimation algorithms v/s SNR for the source at $45°$.

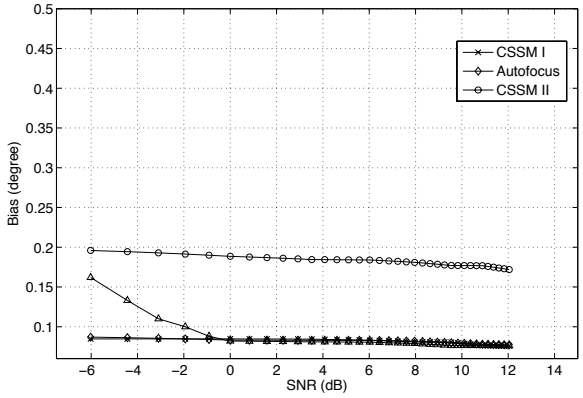


Figure 5.2: Comparison of Bias of the different wideband DOA estimation algorithms v/s SNR for the source at $45°$..

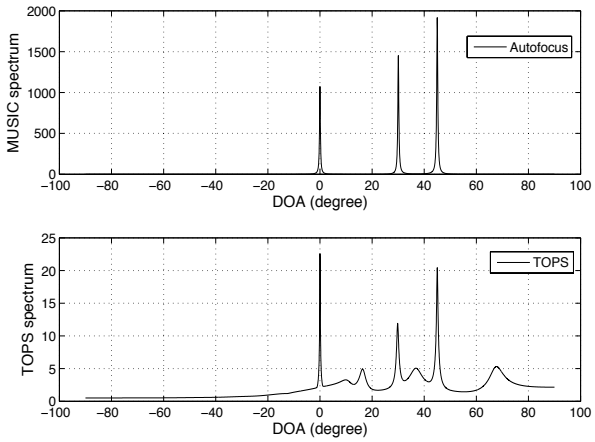


Figure 5.3: MUSIC spectrum for the three sources at 0°, 30° and 45° using i) proposed autofocusing approach (top) and ii) TOPS (bottom) at SNR=0 dB.

the proposed method clearly shows only three peaks, the TOPS spectrum shows several other false peaks, which can potentially get erroneously detected as source DOAs.

5.6 Introduction to Frequency Invariant Beamforming

In this section, we will address a well known problem in wideband array processing, namely frequency invariant beamforming. Unlike narrowband signal models, the beampattern in wideband arrays varies considerably across the bandwidth of the signal and the goal of frequency invariant beamforming is to compensate for this variation. In a one dimensional linear array with N antenna elements, the most general expression for the wideband beampattern is given by

$$B(\omega, \theta) = \sum_{n=0}^{N-1} w_n e^{-j\omega\tau_n(\theta)}. \tag{5.29}$$

where w_n is the weight multiplied to the output of the nth sensor, ω is the temporal frequency of the impinging wideband sources and $\tau_n(\theta)$ is the delay associated with the nth sensor for a source arriving from a direction θ. As the signals are wideband, the phase of the received signal at different sensors (i.e., $e^{-j\omega\tau_n(\theta)}$) are frequency dependent and so in order to realize a desired beampattern,

say, $B_{des}(\theta)$, we need to satisfy

$$\sum_{n=0}^{N-1} w_n e^{-j\omega\tau_n(\theta)} = B_{des}(\theta). \tag{5.30}$$

As we can see, to satisfy the above relation, the weights $\{w_n\}_{n=0}^{N-1}$ need to be functions of frequency (i.e., $w_n(\omega)$). In other words, we need filters $\{w_n(\omega)\}_{n=0}^{N-1}$ following the N sensors, which are typically realized as TDL filters. Frequency invariant beamforming using this time domain filtering approach has been treated extensively in [161, 192, 193, 194] and various interesting filter design techniques like fan filters, and dilated filterbank, have been explored. In [82], however, Ghavami proposed a novel method to perform frequency invariant beamforming using *only one* multiplier following each sensor, by using a rectangular array instead of a linear array. The main idea was to trade temporal processing for an increase in spatial dimension and it was shown how this could be achieved. This method was extended to the case of 3 dimensional cuboidal arrays and adaptive versions of this beamforming technique using beamspace beams, was proposed in [111, 114].

Inspired by the fact that spatial and temporal processing can be interchanged, in this chapter, we seek to explore more general 2 dimensional array geometries and examine if they can be used for frequency invariant beamforming without TDL filter. Specifically, in the following section, we shall consider a more general class of 2 dimensional arrays, viz., arrays with sensors on a lattice, and demonstrate how they provide a general framework for frequency invariant beamforming without using TDL filters. The rectangular array based FIB happens to be a special case of this and the related theory can be readily rederived using our proposed general framework.

5.7 Theory of Lattice Based Frequency Invariant Beamforming without Filters

Lattices provide a useful means to sample a 2 dimensional signal on uniform grid. A special case of a lattice is a rectangular array. For such an array of size $M \times N$ with spacings d_1 and d_2 along the two dimensions respectively, the position vector of the (m, n)th sensor is given by

$$\begin{pmatrix} md_1 \\ nd_2 \end{pmatrix} = \begin{pmatrix} d_1 & 0 \\ 0 & d_2 \end{pmatrix} \begin{pmatrix} m \\ n \end{pmatrix} \tag{5.31}$$

where $m = 0, 1, \cdots, M-1$ and $n = 0, 1, \cdots, N-1$. The lattice generator is $\mathbf{V} = \begin{pmatrix} d_1 & 0 \\ 0 & d_2 \end{pmatrix}$

Two dimensional antenna arrays with sensors on a lattice constitute a major class of frequently used antenna arrays. Consider such an antenna array consisting of MN elements whose elements are on a lattice defined by the generator matrix $\mathbf{V}$. Specifically, the (m, n)th sensor is located at the position vector given by

$$\begin{pmatrix} x_m \\ y_n \end{pmatrix} = \mathbf{V} \begin{pmatrix} m \\ n \end{pmatrix}, m = 0, \cdots, M-1, n = 0, \cdots, N-1$$

where x_m and y_n denote the X-coordinate and the Y-coordinate of the position vector respectively in the $X - Y$ plane. Consider a wideband signal $s^{(c)}(t)$ arriving from an elevation angle ϕ and azimuthal angle θ. The signal received at the (m, n)th antenna, after demodulation of the carrier frequency f_c, is given by

$$x_{m,n}^{(c)}(t) = s^{(c)}(t - \tau_{m,n}) e^{-j2\pi f_c \tau_{m,n}}. \tag{5.32}$$

where

$$\tau_{m,n} = \frac{1}{c} \begin{bmatrix} \sin\phi\cos\theta & \sin\phi\sin\theta \end{bmatrix} \mathbf{V} \begin{bmatrix} m \\ n \end{bmatrix}. \tag{5.33}$$

Here c is the propagation velocity. After sampling at period of T_s (where T_s is less than the Nyquist interval for $s^{(c)}(t)$), the discrete-time Fourier Transform (DTFT) of the (m, n)th sensor, is given by

$$X_{m,n}(e^{j\omega}, \theta) = e^{-j\frac{\widehat{\omega}}{T_s}\tau_{m,n}} S(e^{j\omega}) \tag{5.34}$$

Here $\widehat{\omega} = \omega + 2\pi f_c T_s$. Signal received at (m, n)th sensor is then multiplied by the weight $C_{m,n}$ and then all these signals from MN sensors are summed up to get the output signal

$$Y(e^{j\omega}, \theta) = H(e^{j\omega}, \theta) S(e^{j\omega})$$

where

$$H(e^{j\omega},\theta)=\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}C_{m,n}e^{-j\frac{\tilde{\omega}}{T_s}\tau_{m,n}} \tag{5.35}$$

is the beampattern which is a function of both temporal frequency ω and spatial direction θ. Our aim is to make this beampattern frequency invariant, i.e., all-pass with respect to ω:

$$H(e^{j\omega},\theta)=G(\theta) \tag{5.36}$$

where $G(\theta)$ is a desired narrowband beampattern that we wish to realize. It is to be noted that we keep ϕ fixed in all cases, i.e., we define the beampattern with respect to the angle θ only. This means in particular that we only look into a particular azimuthal angle ϕ. The usual approach to achieve this with a ULA will be to make each of the weights $C_{m,n}$ a function of frequency themselves, i.e., apply a TDL filter at the output of each sensor. However, since we are considering two dimensional arrays, the extra spatial dimension can be used to avoid the temporal dimension, thereby making each $C_{m,n}$ a constant weight instead of TDL filter.

5.7.1 Realization of Lattice Based Frequency Invariant Beams

In this section, we shall develop the theory of realizing a given frequency invariant beampattern (i.e., realize (5.36)) using a two dimensional array with sensors on a lattice. A special case of this using rectangular array was discussed in [82], which we shall generalize to the case of arbitrary, in particular, non separable lattices and study the effect of the lattice geometry qualitatively on the realized beampattern.

In order to realize the frequency invariant beampattern without TDL filters using a 2-D lattice, we can rewrite (5.35) as

$$G(\theta)=\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}C_{m,n}e^{-j\frac{\tilde{\omega}}{T_s}\tau_{m,n}} \tag{5.37}$$

$$=\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}C_{m,n}e^{-j\frac{\tilde{\omega}}{cT_s}\begin{bmatrix}\sin\phi\cos\theta\\ \sin\phi\sin\theta\end{bmatrix}\mathbf{v}\begin{bmatrix}m\\ n\end{bmatrix}} \tag{5.38}$$

Writing $\mathbf{V} = \begin{bmatrix} v_{11} & v_{12} \\ v_{21} & v_{22} \end{bmatrix}$, and assuming $\phi = 90°$, we can simplify (5.38) as

$$G(\theta) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} C_{m,n} e^{-j\frac{\hat{\omega}}{cT_s}[m(v_{11}\cos\theta + v_{21}\sin\theta) + n(v_{12}\cos\theta + v_{22}\sin\theta)]} \tag{5.39}$$

Now, we define two new variables as

$$\omega_1 = \frac{\hat{\omega}}{cT_s}(v_{11}\cos\theta + v_{21}\sin\theta) \tag{5.40}$$

$$\omega_2 = \frac{\hat{\omega}}{cT_s}(v_{12}\cos\theta + v_{22}\sin\theta) \tag{5.41}$$

we can rewrite (5.39) as

$$G(\theta) = G_1(\omega_1, \omega_2) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} C_{m,n} e^{-jm\omega_1} e^{-jn\omega_2} \tag{5.42}$$

Thus $G(\theta)$ can be represented as the 2 dimensional Fourier transform of the weights $C_{m,n}$ in the variables ω_1 and ω_2 and hence, theoretically, we can realize $G(\theta)$ by computing the weights $C_{m,n}$ through inverse DFT as

$$C_{m,n} = \frac{1}{MN} \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} G_1(\omega_{1k}, \omega_{2l}) e^{jm\omega_{1k}} e^{jn\omega_{2l}}, \tag{5.43}$$

where $\omega_{1k} = -\pi + \frac{2\pi k}{M}$, $\omega_{2l} = -\pi + \frac{2\pi l}{N}$

Hence, this technique provides a direct way to compute the weights $C_{m,n}$ and it guarantees to approximate the desired frequency invariant beampattern at the sample points due to the FFT relation. However, the dependence of the nature of the approximation on the lattice geometry deserves further discussion. So, in the next section, we shall study the effect of various lattice geometries on the approximation of the 2 dimensional filter $G_1(\omega_1, \omega_2)$ with respect to the transform variables ω_1 and ω_2.

5.7.2 The Beampattern in the Transform domain: Variation of Contours with lattices

Since the two dimensional filter $G(\omega_1, \omega_2)$ is designed in the (ω_1,ω_2) plane, it is important to study the effect of the lattice geometries in this "transform" domain. The original beampattern $G(\theta)$ is all pass with respect to temporal frequency ω and hence its contours in the (ω, θ) plane are straight lines parallel to the ω-axis. However, the change of variables in (5.40) and (5.41) lead to different contours depending on whether ω or θ is held constant. Moreover, the shapes of such contours depend heavily on the lattice geometries. So it is important to study the effect of lattices on these contours which will in turn reveal its effect on the filter approximation.

5.7.2.1 Constant spatial direction (θ) contours:

Elimination of θ between (5.40) and (5.41) yields the relation of ω_1, ω_2 and θ as

$$\tan\theta = \frac{\omega_2 v_{11} - \omega_1 v_{12}}{\omega_2 v_{21} - \omega_1 v_{22}}$$

Thus, the constant θ contours are straight lines in the $(\omega_1 - \omega_2)$ plane. In other words, along these constant θ lines, the two dimensional filter $G_1(\omega_1, \omega_2)$ varies with the temporal frequency ω. The examples of these constant θ contours are shown in Fig. 5.4 for three different angles ($\psi = 10^\circ$, 45° and 90°) between lattice vectors (the vectors are assumed to be of equal length). Our aim is to make the beampattern all-pass with respect to ω in the range $\omega_1 \leq \omega \leq \omega_2$. This means, in each of these cases, we want the beam to be frequency invariant along the lines shown in Fig. 5.4. The detailed effect of variation of the contours with ψ is elaborated later.

5.7.2.2 Constant temporal frequency (ω) contours:

Elimination of ω between (5.40) and (5.41) yields the relation between ω, ω_1 and ω_2 as

$$||\mathbf{V}^{-T}\mathbf{\Omega}||_2^2 = \left(\frac{\hat{\omega}}{cT_s}\right)^2$$

where $\mathbf{\Omega} = [\omega_1 \quad \omega_2]^T$. These contours describe ellipses centered at $(0, 0)$. Physically, it means that along these ellipses in the (ω_1, ω_2) plane, the temporal frequency ω is constant and the beampattern varies with the spatial angle θ. So for different temporal frequencies ω, different ellipses describe

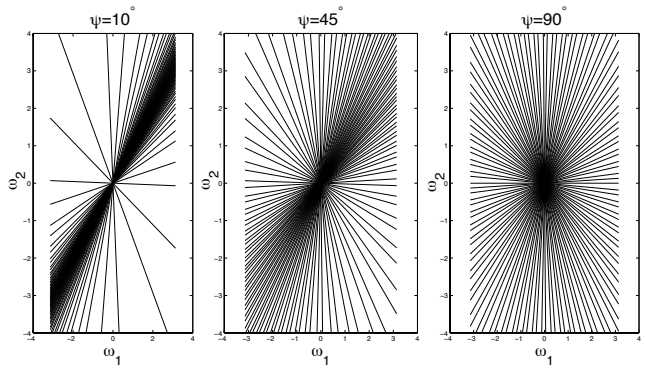


Figure 5.4: Constant $\theta(0 \leq \theta \leq 180)$ contours in $(\omega_1 - \omega_2)$ plane for different values of ψ.

the beampattern as a function of θ and our goal is to make the beam look the same along all these ellipses in the desired range of temporal frequencies. The contours are demonstrated in Fig 5.5 .

For the special case of rectangular array with equal spacing d along the two dimensions ($v_{12} = 0, v_{21} = 0, v_{11} = v_{22} = d$), it can be verified that the ellipses become circles with radii $\frac{\hat{\omega}}{dcT_s}$ as demonstrated in [82].

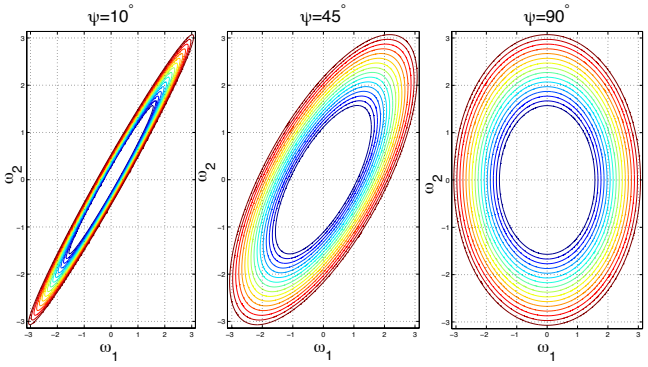


Figure 5.5: Constant ω $(\pi/2 \leq \omega \leq \pi)$ contours in $(\omega_1 - \omega_2)$ plane for different values of ψ.

5.7.2.3 Effect of angle between lattice generator vectors on the beampattern

Given the contours as described earlier, we can now examine the effect of the lattice geometry on the $2D$ frequency response that we want to approximate. In particular, we shall assume the lattice generator to have one of the two generating vectors along the X-axis and the other vector to be

of same length and inclined at an angle ψ and qualify the changes in the contours as the angle ψ varies from 0° to 90°. Then

$$\mathbf{V} = \begin{pmatrix} 1 & \cos\psi \\ 0 & \sin\psi \end{pmatrix}$$

Note that when $\psi = 0$, it degenerates to a ULA and when $\psi = 90^\circ$, it corresponds to a rectangular array. Two properties of $\mathbf{V}$ that are worth mentioning, are:

$$|\mathbf{V}| = \sin\psi$$

and the eigenvalues of $\mathbf{V}^T\mathbf{V}$ are given by

$$\lambda_{1,2} = 1 \pm \cos\psi$$

Now, consider the case of "fixed-θ" contours. For each value of θ, the contour is a straight line making some angle (say ϕ) with the ω_1-axis. Then ϕ, θ and ψ are related by

$$\tan\phi = \sin\psi\tan\theta + \cos\psi$$

In order to observe the effect of variation of ψ on variation in ϕ, the plot of ϕ v/s ψ is given in Fig 5.6.

It can be seen that for $\psi = 0$,

$$\tan\phi = \cos\psi = 1$$

which indicates that the entire beampattern for all values of θ collapses to *one* straight line with slope of 45°. In other words, the frequency response degenerates and the contour collapses to a single line, showing no variation with θ whatsoever. This precisely coincides with the case of a ULA and hence we cannot perform beamforming (with respect to θ) using this method of constant weights per sensor using a ULA (which also agrees with the fact that for a ULA, TDL filters are needed). This can also be alternatively seen from the determinant of $\mathbf{V^TV}$ which gives the volume

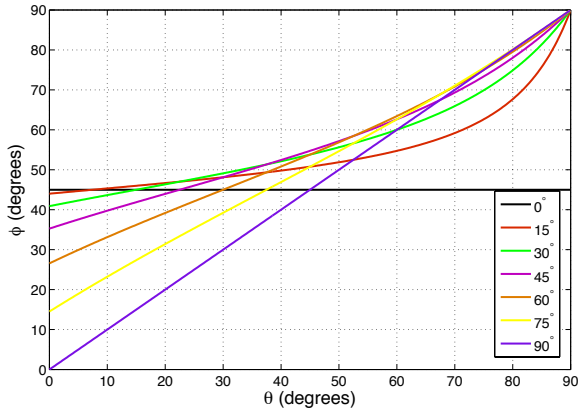


Figure 5.6: Plot of the slope ϕ of the constant θ contours in the $(\omega_1 - \omega_2)$ plane vs the spatial angle θ for different values of the lattice angle ψ.

of the ellipse. For $\psi = 0°$,

$$|\mathbf{V}^T\mathbf{V}| = 0$$

which also indicates that the ellipse collapses to a straight line.

On the other hand, for $\psi = 90°$,

$$\tan\phi = \tan\theta$$

and hence $\phi = \theta$ which indicates that equal increments in the spatial angle θ causes equal increments in the angle ϕ that the corresponding contours make with the axis. Hence the beam is somewhat uniformly distributed along a circular contour in the $(\omega_1 - \omega_2)$ plane as shown in Fig 5.5.

For other values of ψ, particularly for smaller values, ϕ changes more slowly with θ for smaller values of θ and much faster for higher values of θ. Hence, assuming that the mainlobe of the beampattern to be realized is around $\theta = 0°$, a larger change in θ around the mainlobe will cause a *much smaller* change in the slope ϕ of the contours. Hence the mainlobe and adjacent sidelobes are all mapped to a small sector of values of ϕ leading to a *crowding effect of the contours*. For the sidelobes farther away, however, a smaller change in θ causes a much larger change in the value of ϕ and hence the farther sidelobes are sparsely spread out (the angle between the lines increases) as demonstrated in Fig. 5.4.

Next, the effect of the lattice on the constant-ω contours is determined by the eigenvalues of $\mathbf{V}^T\mathbf{V}$. Suppose we are interested in making the beampattern frequency invariant over a range of frequencies $\omega_a \leq \omega \leq \omega_b$. Then this amounts to making $G_1(\omega_1, \omega_2)$ allpass over an annular elliptical region (shown in Fig. 5.5). The width of this annular region, however, varies, being maximum along the major axis and minimum along the minor axis direction. The width is proportional to the eigenvalues and is given by $(\omega_b - \omega_a)(1 + \cos\psi)$ along the major axis and $(\omega_b - \omega_a)(1 - \cos\psi)$ along the minor axis. However, for $\psi = 90^\circ$, the annular region has uniform width.

Hence, smaller the angle ψ between lattice vectors, the desired beampattern $G(\theta)$ gets mapped to the two dimensional filter $G_1(\omega_1, \omega_2)$ such that parts of the beampattern around the mainlobe is mapped to a much smaller sector whereas the farther sidelobes are spread out over a larger sector. However, this crowding effect keeps decreasing with increasing ψ until, at $\psi = 90^\circ$, the beam is uniformly spread over the entire circle.

5.7.3 Realization Issues: Rectangular v/s Circular Sampling

The effect of the lattice geometry on the contours of the 2 dimensional filter $G_1(\omega_1, \omega_2)$ brings us to addressing an important issue: the effect of different sampling geometries in the (ω_1, ω_2) plane on the quality of approximation of the desired beampattern. The basic equation that represents this approximation is (5.42). One way to realize it to compute direct 2 dimensional IDFT of the samples of $G_1(\omega_1, \omega_2)$ in the (ω_1, ω_2) plane and thereby obtain the weights $C_{m,n}$. However, this kind of approach essentially samples the beampattern on a *rectangular grid* in the (ω_1, ω_2) plane at the points $\omega_{1k} = -\pi + \frac{2\pi k}{M}$, $\omega_{2l} = -\pi + \frac{2\pi l}{N}$, $\quad k = 0, 1, \cdots, M-1, \quad l = 0, 1, \cdots, N-1$ However, this kind of sampling in the (ω_1, ω_2) domain does not translate to uniform sampling in the physical (ω, θ) domain, which is desirable if we want to approximate the beampattern uniformly over the range of θ and ω. Due to the effect of different lattice geometries on the contours (as discussed in the previous section), this kind of rectangular sampling results in poorer approximation. For example, it does not distinguish between the regions where the contours are more crowded from the rest. This effect is even more pronounced when ψ is small, since the beam gets more uniformly distributed for higher values of ψ. So the rectangular sampling strategy suits the rectangular array best and its effect worsens as ψ decreases.

In order to overcome this adverse effect of rectangular sampling, we now propose an alternate approach to sampling in the (ω_1, ω_2) plane. Our aim is to approximate the FIB $G(\theta)$ uniformly

over θ and ω. Note that the vector $\mathbf{\Omega} = [\omega_1 \quad \omega_2]^T$ is related to θ and ω through (5.40) and (5.41) and hence sampling uniformly with respect to θ and ω amounts to sampling $\mathbf{\Omega}$ via the following relation:

$$\begin{bmatrix} \omega_{1k,l} \\ \omega_{2k,l} \end{bmatrix} = \frac{\hat{\omega_k}}{cT_s}\mathbf{V}^T \begin{bmatrix} \cos\theta_l \\ \sin\theta_l \end{bmatrix}$$

where $\hat{\omega_k}$ and θ_l are samples of $\hat{\omega}$ and θ uniformly over a rectangular grid, given by,

$$\omega_k = \omega_a + \frac{k}{M}(\omega_b - \omega a), \quad k = 0, 1, \cdots, K-1$$

$$\theta_l = 2\pi\frac{l}{N}, \quad l = 0, 1, \cdots, L-1$$

This is equivalent to sampling in the (ω_1, ω_2) plane *exactly along the elliptical constant-ω and the linear constant-θ contours*. The result of such sampling is illustrated in Fig. 5.7 for $\psi = 10^\circ$ and in Fig. 5.8 for $\psi = 90^\circ$.

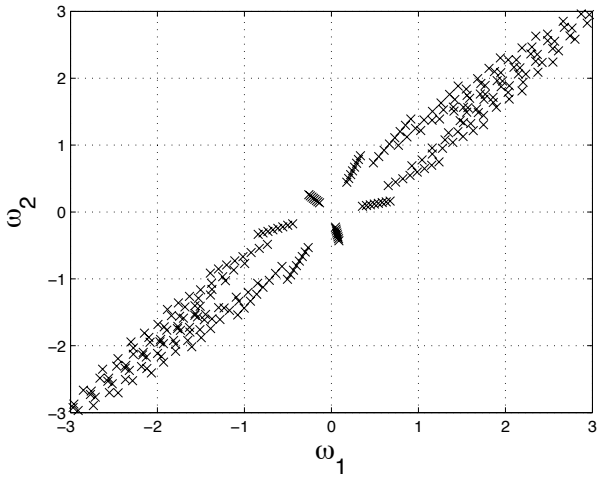


Figure 5.7: Proposed sampling in $(\omega_1 - \omega_2)$ plane for $\psi = 10^\circ$.

Since the sampling is not on a rectangular grid, the IDFT approach cannot be directly applied to compute the weights $C_{m,n}$. Alternately, we can represent (5.42) in the form of a matrix equation and solve for $C_{m,n}$ using least squares by taking Moore Penrose pseudo inverse.

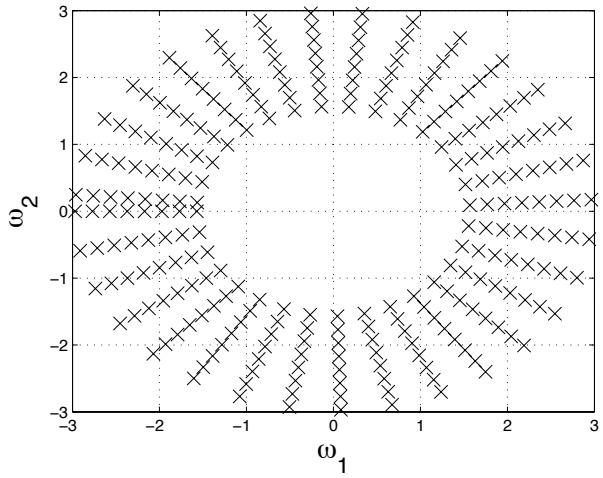


Figure 5.8: Proposed sampling in $(\omega_1 - \omega_2)$ plane for $\psi = 90^\circ$.

5.7.4 Examples

We now provide two design examples for implementing frequency invariant beams using two different lattices, one with lattice generator matrix $\mathbf{V}$ given by (5.44) for two extreme values of ψ, viz., $\psi = 15^\circ$ and $\psi = 90^\circ$. We assume $M = N = 17$ for both examples, and wish to design a simple sinc beampattern with 7 lobes, frequency invariant over the range of $\pi/2 \leq \omega \leq \pi$. We follow the proposed sampling strategy and compute the beamformer weights using least squares. The resultant beampattern for $\psi = 15^\circ$ is plotted in Fig. 5.9 for 50 values of ω in the above mentioned range. The two dimensional plot (as a function of ω and $\sin\theta$) is plotted in Fig. 5.10. The corresponding 1 D and 2 D plots for $\psi = 90^\circ$ are plotted in Fig. 5.11 and Fig.5.12 respectively. It can be clearly seen that under the proposed sampling scheme, the lattice with $\psi = 15^\circ$ outperforms the one with $\psi = 90^\circ$ (rectangular array).

5.7.5 RMSE of approximation

Using the proposed sampling technique, the result of variation of RMSE of approximation of the FIB as a function of the angle between lattice vectors ψ ($\mathbf{V}$ is given by (5.44)) is plotted in Fig. 5.13. Curiously, the simulations indicate that for the proposed sampling strategy, the RMSE is minimum at an angle close to $\psi = 15^\circ$. While the reason for it being so exactly at that angle is difficult to quantify, it can be however, qualitatively explained as follows:

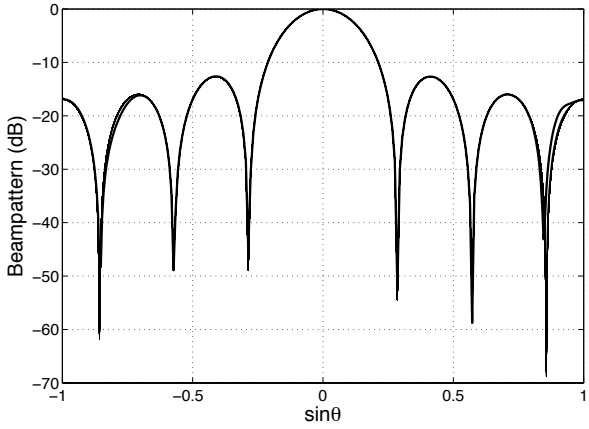


Figure 5.9: Frequency Invariant Beampattern as a function of $\sin\theta$ for 50 values of ω uniformly between $\pi/2$ and π realized using lattice with $\psi = 15^\circ$, $M = N = 17$.

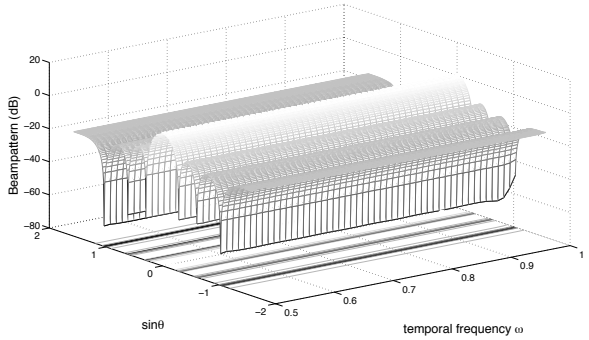


Figure 5.10: 2 dimensional plot of realized frequency invariant beampattern, as a function of spatial direction $\sin\theta$ and temporal frequency ω, realized using a lattice with $\psi = 15^\circ$ and $M = N = 17$.

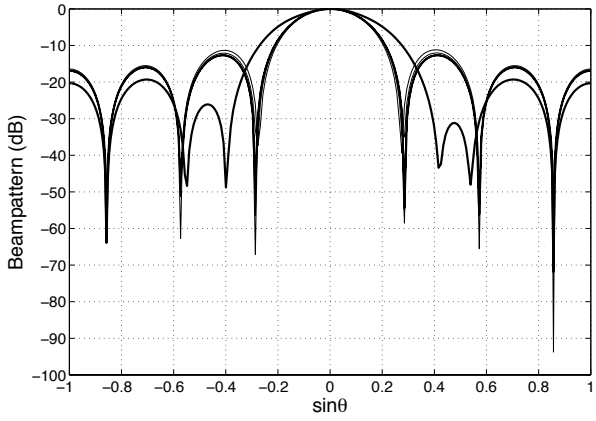


Figure 5.11: Frequency Invariant Beampattern as a function of $\sin\theta$ for 50 values of ω uniformly between $\pi/2$ and π realized using lattice with $\psi = 90^\circ$, $M = N = 17$.

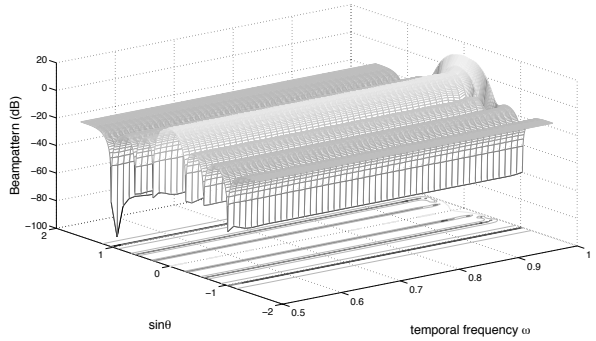


Figure 5.12: 2 dimensional plot of realized frequency invariant beampattern, as a function of spatial direction $\sin\theta$ and temporal frequency ω, realized using a lattice with $\psi = 90^\circ$ and $M = N = 17$.

The uniform sampling with respect to θ and ω leads to sampling of ω_1 and ω_2 along the contours described earlier. And the least squares algorithm then tends to interpolate the values of the beampattern for the points in between the sample points and this interpolation works in the (ω_1, ω_2) plane. For ψ around 90°, the contours are uniformly spread out whereas for lower values of ψ, the contours near the mainlobe region (which has higher magnitude) are crowded together which makes it easier for the least squares to interpolate the intermediate points more efficiently for lower values of ψ. However, for very low values of ψ, (which reduces the lattice to almost an ULA), the two dimensional contours degenerate into a straight line and hence the frequency invariant beamforming strategy using constant weights (instead of TDL filters) fails. So in between these two extremes, there must exist be a value of ψ where the RMSE will hit the minimum and then reverse its trend. This qualitatively explains the observed minimum in the RMSE around 15°. This also shows the important fact that non separable lattices can yield better performance compared to separable (rectangular) lattices in approximating FIB.

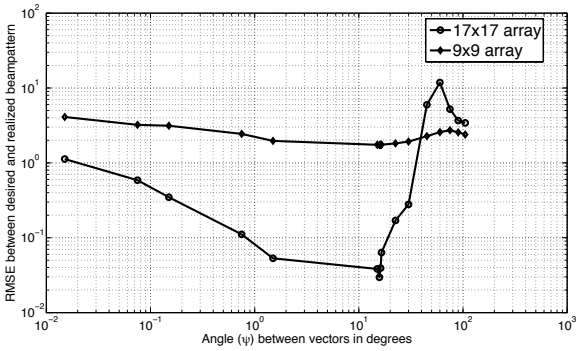


Figure 5.13: RMSE of realizing the frequency invariant beam v/s lattice angle ψ for two arrays of size 9×9 and 17×17.

5.8 Efficient Realization of the lattice based frequency invariant beamforming using Virtual Array

Though the lattice based frequency invariant beamforming proposed in Section 5.7 avoids the need for temporal processing, it results in a proportional increase in the number of physical sensors to achieve this. Precisely, compared to an ULA with N sensors, we require $O(N^2)$ sensors in the 2

dimensional array. So in this section, we propose a novel solution to this problem, by first suggesting a new approach to beamforming based on the "difference co-array" of a physical array and thereby suggesting a way to realize the beampattern generated by a lattice (with $O(N^2)$ virtual sensors) using only $O(N)$ physical sensors. This results from the fact that the geometry of the lattice enables one to represent it virtually as the "cross difference co-array" between two suitable ULAs, and our proposed approach to beamforming can exploit the resulting degrees of freedom of the virtual array, as we shall demonstrate in the following subsections.

5.8.1 Wideband Beamforming Based on Difference Co-array

Extension of the difference co-array based beamforming to the case of wideband signals is unfortunately not straightforward, because now, we will have a different co-array at each frequency. Since our goal is to do frequency invariant beamforming, we propose the following approach to get the effect of the same co-array at each frequency:

Consider D wideband signals, occupying frequency band ω_a to ω_b, impinging on an array of N sensors. Then, the signal received at the nth sensor is given by

$$x_n(t) = \sum_{i=1}^{D} s_i(t - \tau_n(\theta_i))$$

where $\tau_n(\theta_i)$ is the delay associated with the nth array element and the ith source signal and depends on the angle of arrival of the signal and the array geometry. For simplicity, we only consider the elevation angle θ though the delay is also a function of the azimuthal angle. Defining $\mathbf{x}(t) = [x_1(t)x_2(t)\cdots x_N(t)]^T$, and taking the autocorrelation of $\mathbf{x}(t)$, we get

$$\mathbf{R_{xx}}(t) = E(\mathbf{x}(t)\mathbf{x}^H(t)) \tag{5.44}$$

where the (m,n)th element of $\mathbf{R_{xx}}$ is given by

$$\begin{aligned}[\mathbf{R_{xx}}]_{m,n} &= E(x_m(t)x_n^*(t)) \\ &= E\Big[\sum_{i=1}^{D} s_i(t - \tau_m(\theta_i)) \sum_{j=1}^{D} s_j^*(t - \tau_n(\theta_j))\Big]\end{aligned} \tag{5.45}$$

Assuming the sources are uncorrelated (i.e., $E[s_i(t)s_j^*(t-\tau)] = 0, \quad \forall \quad \tau$) we can rewrite (5.45) as

$$[\mathbf{R_{xx}}]_{m,n} = \sum_{i=1}^{D} R_{s_i,s_i}(\tau_n(\theta_i) - \tau_m(\theta_i))$$

where $R_{s_i,s_i}(\tau)$ denotes the autocorrelation of the ith signal. Now, we vectorize $\mathbf{R_{xx}}$ to get the vector $\mathbf{y} \in \mathcal{C}^{N^2\times 1}$. Treating this as the equivalent signal vector (corresponding to the co-array), we wish to apply weights to this vector to perform spatial filtering. Denoting the weight vector as $\mathbf{w} \in \mathcal{C}^{N^2\times 1}$, we index its elements by the double index $(m,n), m = 1,\cdots,N, n = 1,\cdots,N$ for convenience. Then, the final weighted output can be written as

$$\begin{aligned} z &= \mathbf{w}^H\mathbf{y} && (5.46)\\ &= \sum_{i=1}^{D}\sum_{m=1}^{N}\sum_{n=1}^{N} w_{m,n}R_{s_i,s_i}(\tau_n(\theta_i) - \tau_m(\theta_i)) && (5.47)\\ &= \sum_{i=1}^{D}\sum_{m=1}^{N}\sum_{n=1}^{N} w_{m,n}\frac{1}{2\pi}\int_{\omega_a}^{\omega_b} P_{s_i,s_i}(\omega)e^{j\omega(\tau_n(\theta_i)-\tau_m(\theta_i))}\,d\omega && (5.48)\end{aligned}$$

where $P_{s_i,s_i}(\omega)$ denotes the power spectral density of $s_i(t)$. Assuming the sources are bandlimited to $\omega_a \le \omega \le \omega_b$, we can further rewrite (5.48) as

$$\begin{aligned} z &= \frac{1}{2\pi}\sum_{i=1}^{D}\int_{\omega_a}^{\omega_b}\Big(\sum_{m=1}^{N}\sum_{n=1}^{N} w_{m,n}e^{j\omega(\tau_n(\theta_i)-\tau_m(\theta_i))}\Big)P_{s_i,s_i}(\omega)\,d\omega && (5.49)\\ &= \frac{1}{2\pi}\sum_{i=1}^{D}\int_{\omega_a}^{\omega_b} B(\omega,\theta_i)P_{s_i,s_i}(\omega)\,d\omega && (5.50)\end{aligned}$$

where

$$B(\omega,\theta) \triangleq \sum_{m=1}^{N}\sum_{n=1}^{N} w_{m,n}e^{j\omega(\tau_n(\theta)-\tau_m(\theta))}$$

is the beampattern (function of both temporal frequency ω and the spatial angle θ). Our goal is to realize a frequency invariant beampattern, i.e., achieve

$$B(\omega,\theta) = G(\theta), \quad \omega_a \le \omega \le \omega_b \tag{5.51}$$

where $G(\theta)$ is a frequency invariant beampattern (e.g., a narrowband beampattern). In this case, we can rewrite (5.50) as z=$\sum_{i=1}^{D} G(\theta_i)\frac{1}{2\pi}\int_{\omega_a}^{\omega_b} P_{s_i,s_i}(\omega)\,d\omega$

$= \sum_{i=1}^{D} G(\theta_i)\sigma_i^2$ where σ_i^2 denotes the power of the ith source signal.

Taking the additive noise at each antenna also into account, and assuming that it is spatially and temporally white, uncorrelated with the signals, the final output becomes

$$z = \sum_{i=1}^{D} G(\theta_i)\sigma_i^2 + \sigma_n^2 \mathbf{w}^H \vec{\mathbf{1}}_n. \tag{5.52}$$

where σ_n^2 denotes the noise power. It can be noted that (5.52) has the same form as the corresponding narrowband version in (2.7). The technique of making the beampattern frequency invariant, as in (5.51), reduces the wideband beamforming problem essentially to that of the conventional narrowband beamforming, which can be effectively used for detection.

The key step to this approach to frequency invariant beamforming is to realize (5.51). This is the same as requiring

$$\sum_{m=1}^{N}\sum_{n=1}^{N} w_{m,n} e^{j\omega(\tau_n(\theta)-\tau_m(\theta))} = G(\theta). \tag{5.53}$$

In traditional frequency invariant beamforming, the weights w_n are chosen as functions of frequency (i.e., realized as TDL filters) to realize the frequency invariant beampattern $G(\theta)$. However, such filters are realized using $O(N)$ physical sensors arranged in a linear fashion, and the remaining dimension is provided by the temporal delay taps. Another approach to achieve this without performing temporal filtering, is by increasing the physical dimension of the array, using lattices which was discussed in Section 5.7 of this chapter (a special case of which is the rectangular lattice used by Ghavami.) which provides $O(N^2)$ sensors. However, our formulation in this section is based on a nonlinear approach to beamforming (using the second order statistics of the signal), which *automatically provides increase in degrees of freedom* without using extra physical sensors, thereby avoiding the need for temporal filtering. In particular, we shall virtually realize the exact same physical sensors as provided by a $N \times N$ lattice using only $2N$ sensors properly selected (thereby forming a virtual lattice) by exploiting the *cross difference set* of two linear arrays, to which we can apply the lattice based frequency invariant beamforming technique discussed in detail in Section 5.7.

5.8.2 Realization of the lattice as the difference co-array

In this section, we prove a lemma which will enable us to realize a 2 dimensional lattice virtually using only 2 suitably constructed 1 dimensional antenna arrays by using the cross difference co-array between them:

Lemma 5.8.1. *Any 2-dimensional antenna array of size $M \times N$ with elements on a lattice, can be realized using the difference co-array of two uniform linear arrays with M and N elements respectively.* □

Proof. Let us consider a 2-D lattice in the XY plane. Let

$$\begin{pmatrix} r_x \\ r_y \end{pmatrix} = \begin{pmatrix} \mathbf{v_0} & \mathbf{v_1} \end{pmatrix} \begin{pmatrix} m \\ n \end{pmatrix} \tag{5.54}$$

denote the (X, Y) coordinates of the (m, n)th element of the 2-D array. Here the 2×1 vectors $\mathbf{v_0}$ and $\mathbf{v_1}$ denote columns of the lattice generator. Then, the phase of a signal with wavelength λ and propagation vector $\mathbf{u}$, received at the (m, n)th element of the lattice array is given by

$$e^{j\frac{2\pi}{\lambda}\mathbf{u}^T(\mathbf{v_0}m+\mathbf{v_1}n)} \tag{5.55}$$

Now let us consider two linear arrays with M antennas and N antennas respectively. The M antennas of the first array are arranged in the form of a ULA, which lies along the same direction in the (XY) plane, as the vector $\mathbf{v_0}$. In other words, the position vector of the mth element of this ULA is

$$\vec{\rho}_{\mathbf{1,m}} = m\mathbf{v_0}, \quad m = 0, 1 \cdots M-1. \tag{5.56}$$

The N antennas of the other array are also arranged in the form of a ULA, which lies along the direction of the vector $-\mathbf{v_1}$. Hence the position vector of the nth element of this second array is

$$\vec{\rho}_{\mathbf{2,n}} = -n\mathbf{v_1}, \quad n = 0, 1 \cdots N-1. \tag{5.57}$$

The cross difference co-array of these two arrays is defined as the array with position vectors $\vec{\mathbf{p}}_{m,n} = \vec{\rho}_{\mathbf{1,m}} - \vec{\rho}_{\mathbf{2,n}}$

$m = 0, 1, \cdots, M-1, n = 0, 1, \cdots, N-1$ which can be readily seen as

$$\vec{\mathbf{p}}_{m,n} = m\mathbf{v_0} + n\mathbf{v_1} \tag{5.58}$$

which corresponds to the (m, n) th sensor of the lattice.

□

Fig. 5.14 shows examples where two $1D$ arrays placed along the columns of the lattice generator matrix give rise to a $2D$ cross difference co-array with elements on the lattice.

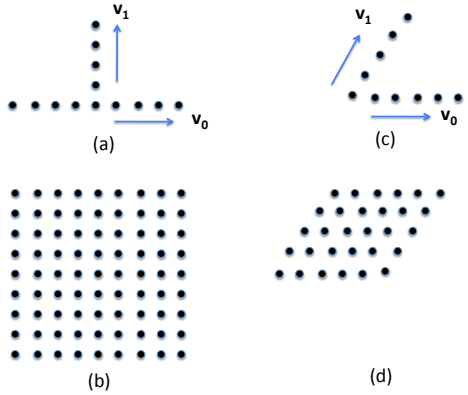


Figure 5.14: Two examples of $1D$ arrays and their cross difference co-array on a $2D$ lattice. (a) and (c) denote the physical arrays, (b) and (d) denote the corresponding cross difference co-arrays.

5.8.3 Frequency Invariant Beamforming using the difference array:

To physically realize the difference co-array corresponding to the lattice, we follow the covariance matrix based beamforming approach discussed earlier. However, instead of considering the *self-difference* co-array, we now consider the *cross difference* co-array of the two ULAs described in the preceding lemma. Let $\mathbf{x_1}(t) \in \mathcal{C}^{M\times 1}$ and $\mathbf{x_2}(t) \in \mathcal{C}^{N\times 1}$ denote the signal vectors received at the M element and N element ULA oriented along $-\mathbf{v_0}$ and $\mathbf{v_1}$ respectively. So, the mth element of $\mathbf{x_1}(t)$ is given by

$$[\mathbf{x_1}]_m(t) = \sum_{i=1}^{D} s_i(t + m\mathbf{u}(\theta_i)^T\mathbf{v_0}) + n_m(t), \quad m = 0, 1, \cdots, M-1$$

where $\mathbf{u}(\theta) = \frac{1}{c}[\cos\theta \quad \sin\theta \quad 0]^T$ denotes the propagation vector assuming θ is the azimuthal angle and $\phi = 90^\circ$ is the elevation angle. The noise term $n_m(t)$ is assumed white Gaussian noise which is spatially and temporally uncorrelated with same power for all the antennas in both arrays. Similarly, the nth element of $\mathbf{x_2}(t)$ is given by

$$[\mathbf{x_2}]_n(t) = \sum_{i=1}^{D} s_i(t - n\mathbf{u}(\theta_i)^T\mathbf{v_1}) + n_n(t), \quad n = 0, 1, \cdots, N-1$$

Next, we take the cross correlation of the signals received at the two ULAs, i.e.,

$$\mathbf{R}_{\mathbf{x_1},\mathbf{x_2}} = E\Big[\mathbf{x_1}(t)\mathbf{x_2}^H(t)\Big]$$

The (m,n)th entry of the cross correlation matrix is given by

$$[\mathbf{R}_{\mathbf{x_1},\mathbf{x_2}}]_{m,n} = \sum_{i=1}^{D}\sum_{j=1}^{D} E(s_i(t + m\mathbf{u}(\theta_i)^T\mathbf{v_0})s_j^*(t - n\mathbf{u}(\theta_j)^T\mathbf{v_1})) + \sigma_n^2\delta(m-n) \tag{5.59}$$

$$= \sum_{i=1}^{D} R_{s_i,s_i}(n\mathbf{u}(\theta_i)^T\mathbf{v_1} + m\mathbf{u}(\theta_i)^T\mathbf{v_0}) + \sigma_n^2\delta(m-n) \tag{5.60}$$

where $R_{s_i,s_i}(\tau)$ denotes the autocorrelation of the ith source signal at lag τ and (5.60) results because of the fact that the ith and jth signals are uncorrelated for $i \neq j$. We can rewrite (5.60) in terms of the power spectral density as

$$[\mathbf{R}_{\mathbf{x_1},\mathbf{x_2}}]_{m,n} = \sum_{i=1}^{D}\int_{\omega_a}^{\omega_b} S_{i,i}(\omega)e^{j\omega(n\mathbf{u}(\theta_i)^T\mathbf{v_1} + m\mathbf{u}(\theta_i)^T\mathbf{v_0})}d\omega + \sigma_n^2(m-n)$$

where $S_{i,i}(\omega)$ is the power spectral density of the ith wideband source. Now, vectorizing $\mathbf{R}_{\mathbf{x_1},\mathbf{x_2}}$, we get the vector $\mathbf{z}$, given by:

$$\mathbf{z} = \int_{\omega_a}^{\omega_b} \mathbf{A}(\omega, \theta_1, \cdots, \theta_D)\mathbf{S}(\omega)d\omega + \sigma_n^2\vec{\mathbf{1}}_n$$

where $\mathbf{A}(\omega, \theta_1, \theta_2, \cdots, \theta_D) = [\mathbf{a}(\omega, \theta_\mathbf{1}) \quad \mathbf{a}(\omega, \theta_\mathbf{2}) \cdots \mathbf{a}(\omega, \theta_\mathbf{D})] \in \mathcal{C}^{MN\times D}$. For convenience, we denote the elements of $\mathbf{a}(\omega, \theta_i) \in \mathcal{C}^{MN\times 1}$ by the double index $(m,n), m = 0, 1, \cdots, M-1, n = 0, 1, \cdots, N-1$ as

$$[\mathbf{a}(\omega, \theta_i)]_{m,n} = e^{j\omega(n\mathbf{u}(\theta_i)^T\mathbf{v_1} + m\mathbf{u}(\theta_i)^T\mathbf{v_0})}$$

The vector $\mathbf{S}(\omega) \in \mathcal{C}^{D\times 1}$ is given by

$$\mathbf{S}(\omega) = [S_{1,1}(\omega) \quad S_{2,2}(\omega) \cdots S_{D,D}(\omega)]^T$$

Now, we wish to apply constant weight vector $\mathbf{w}$ to $\mathbf{z}$ in order to realize a given frequency invariant beampattern. Then the output of this beamformer is given by

$$y = \sum_{i=1}^{D} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} w_{m,n} e^{j\omega(n\mathbf{u}(\theta_i)^T\mathbf{v_1} + m\mathbf{u}(\theta_i)^T\mathbf{v_0})} S_{i,i}(\omega)$$

For frequency invariant beamforming, we need

$$\sum_{m=0}^{M-1} \sum_{n=0}^{N-1} w_{m,n} e^{j\omega(n\mathbf{u}(\theta_i)^T\mathbf{v_1} + m\mathbf{u}(\theta_i)^T\mathbf{v_0})} = G(\theta)$$

where $G(\theta)$ is the desired frequency invariant beampattern. It can be readily observed that the above relation is equivalent to realizing (5.39) and hence the methods discussed in Section 5.7 can be used to realize this frequency invariant beampattern.

This way, we demonstrated, that using only $2N$ physical sensors and performing beamforming with respect to the difference co-array, we can indeed avoid TDL filters and yet realize frequency invariant beams without increasing the number of sensors.

5.8.4 Examples

We now provide design examples to validate the proposed virtual implementation of the lattice. We wish to implement a lattice with generator matrix given by (5.44) with $\psi = 45^\circ$ with $M = N = 17$. Hence we use two 17 element ULAs at an angle of 45° with each other and perform the proposed covariance matrix based beamforming. To test the beampattern, we vary the direction of arrival of the impinging signal and plot the corresponding output of the beamformer. In order to test the frequency invariance of the beamformer, we transmit three narrowband signals with frequencies $\omega = \pi/2, 3\pi/4$ and π and plot the resulting beampatterns in Fig. 5.15 using 200 snapshots to realize the covariance matrix. As can be seen, the beamformer output is fairly constant for all the three frequencies.

We provide another example as an application of this frequency invariant beamforming in suppressing wideband jammers. We assume the same array structure as in the previous example.

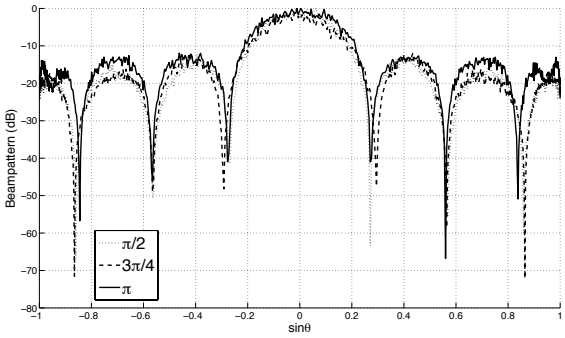


Figure 5.15: Difference co-array based frequency invariant beampattern as a function of $\sin\theta$ using two 17 element ULAs plotted for three values of $\omega = \pi/2, 3\pi/4$ and π, using 200 snapshots.

However, to suppress jammers, we first realize a set of $P = 7$ orthogonal frequency invariant beams spanning the beamspace, and then combine them by using suitable (narrowband) weights in the beamspace, to get the final output. The corresponding beamformer output as it evolves across various snapshots (T), is plotted in Fig. 5.16. The jammer directions are assumed to be $\{60°, 45°, 30°, 15°\}$. It can be seen that for $T = 300$ snapshots, clear nulls have been introduced in the beamformer output in the jammer directions.

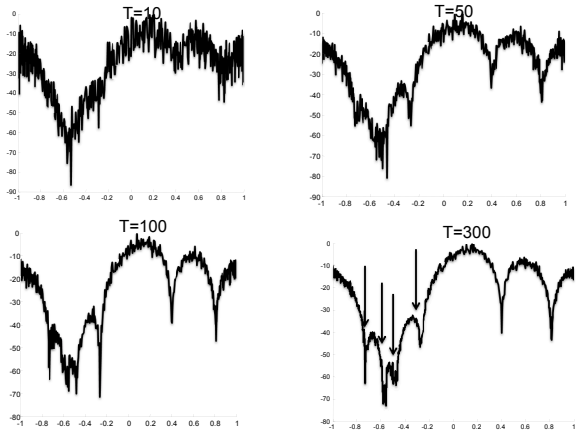


Figure 5.16: Evolution of jammer nulling with snapshots (T) based on the difference co-array using two 17 element ULAs with $\psi = 45°$ with jammer directions given by $\{60°, 45°, 30°, 15°\}$.

5.9 Concluding Remarks

In this chapter, we proposed a new wideband DOA estimation algorithm based on the coherent signal subspace approach. It provides a solution to one of the biggest drawback of coherent methods, viz., the requirement of initial DOA estimates, while exhibiting the desirable properties of coherent methods. The focusing matrices proposed by our method are constructed in a fully adaptive manner, directly from the received data, thereby avoiding the need for careful calibration. Also it results in no SNR loss as the matrices are unitary by construction. However, the performance of our method is likely to depend on the choice of the focusing frequency (subspace) and future work will concentrate on the selection of the best focusing frequency based on the received data. It will also be interesting to perform asymptotic performance analysis of the proposed algorithm and compare the results analytically with the existing algorithms.

The theory of frequency invariant beamforming for wideband signals using two dimensional arrays with sensors on a (nonseparable) lattice was also developed, which avoids the need for any temporal processing by exploiting the increased spatial degrees of freedom. The need for a corresponding increase in the number of physical sensors is further avoided by exploiting the additional degrees of freedom provided by the difference co-array of a physical array. This results in significant reduction in the number of sensors needed, yet achieving the desired performance. The simulation examples satisfactorily show the effectiveness of the proposed technique and indicate that the proposed methods are a better alternative to the conventional wideband sensor array processing.

Chapter 6

Coprime Sampling Arrays and Commuting Coprime Matrices

6.1 Introduction

In this chapter, we will study the family of commuting and coprime integer matrices for efficient sampling of multidimensional signals. Coprime integer matrices which commute under multiplication arise in the theory of multidimensional multirate systems [97], [37], [72], [73], [102], [116]. For example consider the cascade of the multidimensional decimator and expander [182] shown in Fig. 6.1(a), where the decimation matrix $\mathbf{M}$ and the expansion matrix $\mathbf{N}$ are integer matrices. It is well-known (e.g., see [37]) that these two blocks can be interchanged in cascade, that is, Figs. 6.1(a) and 6.1(b) are equivalent, if and only if the matrices $\mathbf{M}$ and $\mathbf{N}$ are commuting and coprime.

More recently, such matrix pairs have been shown to have applications in multidimensional array processing [183], [185]. To explain this, consider two sensor arrays, with sensors located on lattices generated by the integer matrices $\mathbf{M}$ and $\mathbf{N}$ as shown in Fig. 6.2(a), (b). Thus the sensor locations are at $\mathbf{M}\mathbf{n}_1$ and $\mathbf{N}\mathbf{n}_2$ where $\mathbf{n}_1$ and $\mathbf{n}_2$ are integer vectors. Since the density of sensors on the lattices are $1/\det \mathbf{M}$ and $1/\det \mathbf{N}$ respectively [96], [182], the two sets of sensors are distributed sparsely (with respect to the grid of all integers) in space. It can be shown, however, that the difference set $\mathbf{M}\mathbf{n}_1 - \mathbf{N}\mathbf{n}_2$ can take all integer values in the 2D plane (as $\mathbf{n}_1$ and $\mathbf{n}_2$ vary over all integer vectors), if the lattice generators $\mathbf{M}$ and $\mathbf{N}$ are commuting coprime matrices [185]. The importance of this result arises from the fact that the above difference set (also known as the difference coarray [89]) is the set of lags at which the 2D autocorrelation of an impinging signal can be estimated. The fact that *sparse* coprime arrays allow one to estimate correlations on a *dense*

grid has important implications in array processing, similar to the one-dimensional case [184]; one example is that we can identify many more sources than the number of sensors. For more details, the reader should refer to [184], [183], [136], and [185].

The mathematical problem of generating families of commuting coprime integer matrices is therefore of interest to the signal processing community, and this is the main focus of this chapter. The problem in its general form is still open even for the 2×2 case, but useful special cases of this problem have been addressed in the past [97], [102], [72]. In this chapter we enrich the class of known coprime pairs of matrices by presenting several classes of 2×2 matrices: circulants, skew circulants, and triangular matrices. We also consider a specific case of 3×3 matrices, namely triangular matrices and their adjugates. For each of these families we derive necessary and sufficient conditions for coprimality. Of these, the circulant and skew circulant families are automatically commuting families. For the remaining classes, commutativity has to be imposed by further constraining the matrix elements. For each of the families, the special case where the two matrices are adjugates of each other is also elaborated. These are of interest because adjugate pairs have the desirable property that they automatically commute. It will also be shown that any 2×2 matrix can be triangularized using a simple unimodular transformation, and the test for coprimality of triangular pairs of matrices developed in this chapter can be used to test coprimality of arbitrary pairs of integer matrices. For this reason, our result on coprimality of triangular matrices is especially useful.

The content of this chapter is mainly drawn from [137].

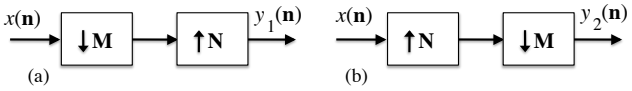


Figure 6.1: (a) A multidimensional decimator and expander in cascade. (b) System with decimator and expander interchanged.

6.1.1 Preliminaries

For convenience we summarize here a number of basic definitions and properties pertaining to integer matrices.

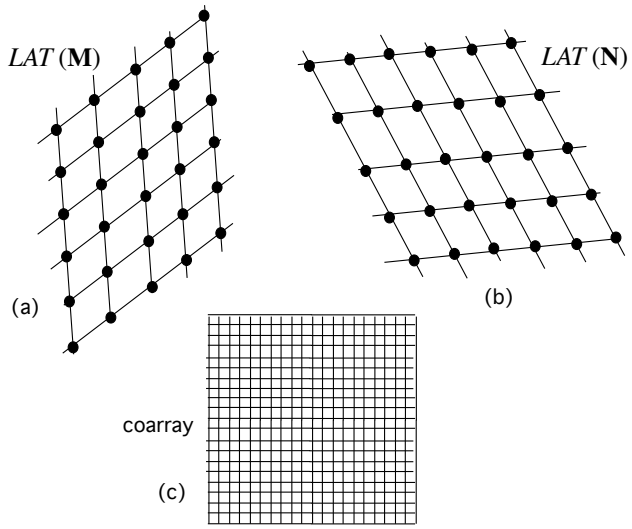


Figure 6.2: (a), (b) Two dimensional arrays with sensors on integer lattices, and (c) the coarray.

1. *Common divisors*. An integer matrix $\mathbf{R}$ is said to be a right divisor of $\mathbf{M}$ if $\mathbf{M} = \mathbf{M}_1\mathbf{R}$ for some integer matrix $\mathbf{M}_1$. Thus $\mathbf{R}$ is a *right common divisor (rcd)* of $\mathbf{M}$ and $\mathbf{N}$ if $\mathbf{M} = \mathbf{M}_1\mathbf{R}$ and $\mathbf{N} = \mathbf{N}_1\mathbf{R}$ for some integer matrices $\mathbf{M}_1$ and $\mathbf{N}_1$. We say that $\mathbf{R}$ is a *greatest right common divisor* (grcd) of $\mathbf{M}$ and $\mathbf{N}$ if, for any other rcd $\mathbf{R}_1$, we can write $\mathbf{R} = \mathbf{P}_1\mathbf{R}_1$ for appropriate integer matrix $\mathbf{P}_1$. The existence of grcds can readily be proved (e.g., see Appendix A.A of [185]).

2. *Unimodularity*. An integer matrix $\mathbf{U}$ with unit determinant is said to be *unimodular* (with $\det\mathbf{U} = -1$ allowed sometimes for convenience). If $\mathbf{R}$ is a grcd then so is $\mathbf{UR}$ for any unimodular $\mathbf{U}$.

3. *Coprimality*. We say that $\mathbf{M}$ and $\mathbf{N}$ are *right coprime* if they have no rcds other than unimodular matrices. When $\mathbf{R}$ is a grcd of $\mathbf{M}$ and $\mathbf{N}$ we can write $\mathbf{M} = \mathbf{M}_1\mathbf{R}$ and $\mathbf{N} = \mathbf{N}_1\mathbf{R}$ where $\mathbf{M}_1$ and $\mathbf{N}_1$ are right coprime. Left common divisors (lcd), greatest left common divisors (glcd), and left coprimality are defined similarly.

4. *Commutativity*. When $\mathbf{M}$ and $\mathbf{N}$ commute, i.e.,

$$\mathbf{MN} = \mathbf{NM}, \tag{6.1}$$

left and right coprimality are equivalent (see [37]).

5. *Bezout's theorem.* According to Bezout's theorem $\mathbf{M}$ and $\mathbf{N}$ are left coprime if and only if there exist integer matrices $\mathbf{P}_1$ and $\mathbf{Q}_1$ such that

$$\mathbf{M}\mathbf{P}_1 + \mathbf{N}\mathbf{Q}_1 = \mathbf{I} \tag{6.2}$$

This equation is called *Bezout's identity*. Similarly, they are right coprime if and only if there exist integer matrices $\mathbf{P}_2$ and $\mathbf{Q}_2$ such that $\mathbf{P}_2\mathbf{M} + \mathbf{Q}_2\mathbf{N} = \mathbf{I}$. For scalar integers, Bezout's theorem is referred to as Euclid's theorem [101], [127]. References for further details include [37], [96], [80], [119], and [182].

6. *Integers on a row or column.* The left coprimality condition (6.2) implies that the gcd of the $2D$ integers in any row of the matrix $[\,\mathbf{M} \quad \mathbf{N}\,]$ is required to be unity. To see this, assume that the integers in the ith row of $[\,\mathbf{M} \quad \mathbf{N}\,]$ have gcd g. Now premultiply both sides of (6.2) by the unit vector $[0\,0\,\ldots 1\,\ldots 0]$ where the 1 appears in the ith position. Then g is a factor on the left side but not on the right, which is a contradiction. Similarly if $\mathbf{M}$ and $\mathbf{N}$ are *right* coprime, the $2D$ integers on any *column* of the matrix $[\,\mathbf{M}^T \quad \mathbf{N}^T\,]^T$ have gcd equal to unity.

Throughout the sequel, the notation

$$(A_1, A_2, \cdots, A_n) \tag{6.3}$$

denotes the greatest common divisor (gcd) of the scalar integers $A_1, A_2, \cdots, A_n$. The following simple fact will be useful in some of the derivations:

Fact . For any two integers A and B, we have

$$(A, B) = (A + B\alpha, B) \tag{6.4}$$

where α is an arbitrary integer. $\diamond$

Proof. Suppose g is a common factor of A and B, so that $A = gX$ and $B = gY$ for integers X, Y. Then $A + B\alpha = g(X + Y\alpha)$ which shows that g is a common factor of $A + B\alpha$ and B. Conversely let h be a common factor of $A + B\alpha$ and B, so $A + B\alpha = hU$ and $B = hV$ for integers U, V. Then $A = h(U - V\alpha)$, so h is also a factor of A and B. Thus every common factor between A and B is also a common factor between $A + B\alpha$ and B, and vice versa. In particular therefore,

$(A, B) = (A + B\alpha, B)$. □

It follows similarly that $(A, B, C, D, \ldots) = (A + B\alpha, B, C, D, \ldots)$.

6.1.2 Chapter outline

Secs. 6.2–6.5 consider 2×2 matrices. Secs. 6.2 and 6.3 discuss circulant and skew circulant pairs of coprime matrices. Triangular matrices are considered in Sec. 6.4. In Sec. 6.5 we introduce the use of simple unimodular transforms to derive further families of coprime matrix pairs. In Sec. 6.5, we also show that the coprimality of any pair of 2×2 matrices can be tested by testing the coprimality of a pair of triangular martrices. Section 6.6 considers coprimality of a 3×3 triangular matrix and its adjugate. Conlcuding remarks are made in Sec. 6.7.

6.2 Circulant matrices

We say that a matrix $\mathbf{P}$ is circulant if each row can be obtained from the preceding row by a right circular shift. In this section we consider pairs of 2×2 circulant matrices:

$$\mathbf{P} = \begin{bmatrix} p & q \\ q & p \end{bmatrix}, \quad \mathbf{P}_1 = \begin{bmatrix} p_1 & q_1 \\ q_1 & p_1 \end{bmatrix} \tag{6.5}$$

It is readily verified that these matrices commute, so "coprimality" means left as well as right coprimality.

First we make some preliminary observations. We know that these matrices are (left) coprime if and only if $\mathbf{PA} + \mathbf{P}_1\mathbf{A}_1 = \mathbf{I}$ for some integer matrices $\mathbf{A}, \mathbf{A}_1$. Reading out the first columns of $\mathbf{A}$ and $\mathbf{A}_1$, we get

$$\begin{bmatrix} p & q \\ q & p \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} + \begin{bmatrix} p_1 & q_1 \\ q_1 & p_1 \end{bmatrix} \begin{bmatrix} a_1 \\ b_1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{6.6}$$

By rearrangement of rows and columns we see that this equation is equivalent to

$$\begin{bmatrix} p & q \\ q & p \end{bmatrix} \begin{bmatrix} b \\ a \end{bmatrix} + \begin{bmatrix} p_1 & q_1 \\ q_1 & p_1 \end{bmatrix} \begin{bmatrix} b_1 \\ a_1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{6.7}$$

Thus, Eq. (6.6) is in fact equivalent to Bezout's idenity:

$$\underbrace{\begin{bmatrix} p & q \\ q & p \end{bmatrix}}_{\mathbf{P}} \underbrace{\begin{bmatrix} a & b \\ b & a \end{bmatrix}}_{\mathbf{A}} + \underbrace{\begin{bmatrix} p_1 & q_1 \\ q_1 & p_1 \end{bmatrix}}_{\mathbf{P}_1} \underbrace{\begin{bmatrix} a_1 & b_1 \\ b_1 & a_1 \end{bmatrix}}_{\mathbf{A}_1} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{6.8}$$

In short, the two circulants $\mathbf{P}$ and $\mathbf{P}_1$ are coprime if and only if Bezout's identity can, in particular, be satisfied with *circulant* matrices $\mathbf{A}$ and $\mathbf{A}_1$. So $\mathbf{P}$ and $\mathbf{P}_1$ are coprime if and only if (6.6) can be satisfied for integers a, b, a_1, b_1. We now prove the following:

Theorem . The circulant matrices $\mathbf{P}$ and $\mathbf{P}_1$ are coprime if and only if

$$(p+q,\ p_1+q_1) = 1 \quad \text{and} \quad (p-q,\ p_1-q_1) = 1, \tag{6.9}$$

i.e., if and only if $p+q$ is coprime to p_1+q_1 and $p-q$ coprime to p_1-q_1. $\diamond$

Proof. The coprimality condition (6.6) can be written as

$$pa + qb + p_1a_1 + q_1b_1 = 1 \tag{6.10}$$

$$qa + pb + q_1a_1 + p_1b_1 = 0 \tag{6.11}$$

Adding these equations results in

$$(p+q)(a+b) + (p_1+q_1)(a_1+b_1) = 1 \tag{6.12}$$

So $\mathbf{P}$ and $\mathbf{P}_1$ are coprime if and only if there exist integers a, b, a_1, b_1 to satisfy (6.11) and (6.12). From (6.12) it is clear that $(p+q)$ and (p_1+q_1) must be coprime (otherwise their gcd would appear on the right). Similarly $(p-q)$ and (p_1-q_1) must be coprime (as seen by subtracting (6.10) from (6.11).

Conversely, assume (6.9) holds. First, since $(p+q)$ and (p_1+q_1) are coprime, there exist integers c and c_1 such that $(p+q)c + (p_1+q_1)c_1 = 1$ (Euclid's theorem [101], [127]). Writing $c = a+b$ and $c_1 = a_1+b_1$, we satisfy (6.12); we can assume that a and a_1 are still free, since b and b_1 can be adjusted to yield the right values for c and c_1. Now, Eq. (6.11) can be rewritten in terms of c and c_1

as $qa + p(c-a) + q_1 a_1 + p_1(c_1 - a_1) = 0$ which can be rearranged as

$$(p-q)a + (p_1 - q_1)a_1 = pc + p_1 c_1 \tag{6.13}$$

The right hand side above is fixed. Since $(p-q)$ and $(p_1 - q_1)$ are coprime, we can choose the free integers a and a_1 such that (6.13) holds. We then solve for $b = c - a$ and $b_1 = c_1 - a_1$. So there exist a, b, a_1, b_1 satisfying (6.11) and (6.12) (hence (6.10) and (6.11)). □

6.2.1 Relation to DFT

Notice that $p+q$ and $p-q$ are the two DFT coefficients of the sequence $\{p, q\}$. For further insight into Theorem , recall here that any circulant matrix has eigenvalues equal to the DFT coefficients of its first row, and eigenvectors equal to the columns of the DFT matrix [146], [182]. Thus, circulants are diagonalized by the DFT matrix $\mathbf{W}$:

$$\underbrace{\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}}_{\mathbf{W}} \begin{bmatrix} p & q \\ q & p \end{bmatrix} \underbrace{\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}}_{\mathbf{W}^\dagger} = 2 \begin{bmatrix} p+q & 0 \\ 0 & p-q \end{bmatrix} \tag{6.14}$$

Theorem simply says that the DFT coefficients arising from the circulant matrix $\mathbf{P}$ must be coprime to the corresponding DFT coefficients arising from the circulant matrix $\mathbf{P}_1$. A second proof of the theorem, based only on DFT, is given next, for further insight.

Second proof of Theorem . We know $\mathbf{P}$ and $\mathbf{P}_1$ are left coprime if and only if Eq. (6.8) holds for some circulants $\mathbf{A}$ and $\mathbf{B}$. Since $\mathbf{W}\mathbf{W}^\dagger = 2\mathbf{I}$ we can rewrite Eq. (6.8) as $\mathbf{W}\mathbf{P}\mathbf{A}\mathbf{W}^\dagger + \mathbf{W}\mathbf{P}_1\mathbf{A}_1\mathbf{W}^\dagger = 2\mathbf{I}$ or equivalently as

$$\mathbf{W}\mathbf{P}\mathbf{W}^\dagger\mathbf{W}\mathbf{A}\mathbf{W}^\dagger + \mathbf{W}\mathbf{P}_1\mathbf{W}^\dagger\mathbf{W}\mathbf{A}_1\mathbf{W}^\dagger = 4\mathbf{I} \tag{6.15}$$

Since $\mathbf{P}, \mathbf{A}, \mathbf{P}_1$, and $\mathbf{A}_1$ are circulants, they can be diagonalized as in Eq. (6.14). So (6.15) is equivalent to

$$\begin{bmatrix} p+q & 0 \\ 0 & p-q \end{bmatrix} \begin{bmatrix} a+b & 0 \\ 0 & a-b \end{bmatrix}$$

$$+\begin{bmatrix} p_1+q_1 & 0 \\ 0 & p_1-q_1 \end{bmatrix}\begin{bmatrix} a_1+b_1 & 0 \\ 0 & a_1-b_1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{6.16}$$

Summarizing, $\mathbf{P}$ and $\mathbf{P}_1$ are coprime if and only if there exist integers a, b, a_1, b_1 such that

$$(p+q)(a+b)+(p_1+q_1)(a_1+b_1)=1 \tag{6.17}$$

$$(p-q)(a-b)+(p_1-q_1)(a_1-b_1)=1 \tag{6.18}$$

These imply the condition given by Eq. (6.9) indeed. Conversely, if (6.9) holds then there exist integers c, c_1, d, d_1 such that

$$(p+q)c+(p_1+q_1)c_1=1 \tag{6.19}$$

$$(p-q)d+(p_1-q_1)d_1=1 \tag{6.20}$$

Setting $c=(a+b), d=(a-b), c_1=a_1+b_1$, and $d_1=a_1-b_1$, we can then solve for $a=(c+d)/2, b=(c-d)/2, a_1=(c_1+d_1)/2$, $b_1=(c_1-d_1)/2$. To complete the proof that $\mathbf{P}$ and $\mathbf{P}_1$ are coprime, we now argue that c, d (and similarly c_1, d_1) can be assumed to have the same parity (both even or both odd), so that a, b, a_1, b_1 turn out to be integers.

The solutions c, c_1 and d, d_1 in Eqs. (6.19), (6.20) are not unique. For example, starting from some solution if we make the replacements

$$c \to c+(p_1+q_1), \quad c_1 \to c_1-(p+q) \tag{6.21}$$

$$d \to d+(p_1-q_1), \quad d_1 \to d_1-(p-q) \tag{6.22}$$

the result will continue to be a solution. We provide here the (somewhat boring) argument that if the parities of c and d (and similarly c_1 and d_1) do not match, then such replacements can always be used to match up parities. First consider Eq. (6.19). Since the right side is unity, the two terms in the left have opposite parity. Without loss of generality assume the 1st term is even. Then there are two cases:

1. $(p+q)$ *is odd.* In this case c is even, c_1 is odd, and (p_1+q_1) is odd. So in the second equation Eq. (6.20), $(p-q)$ and (p_1-q_1) are odd. Since one term in (6.20) is even and the other odd, there are now two possibilities: (a) d is even and d_1 is odd, and (b) d is odd and d_1 is even. In

the first event, c, d have the same parity, and so do c_1, d_1, so all is well. In the second event if we perform the replacement (6.22) then the parities of d and d_1 are changed (because $(p-q)$ and (p_1-q_1) are odd), and once again c, d have the same parity, and so do c_1, d_1.

2. $(p+q)$ *is even.* In this case c can be of any parity, c_1 is odd, and (p_1+q_1) is odd. So $(p-q)$ is even and (p_1-q_1) is odd. In order for (6.20) to be true, d_1 is therefore odd, and d can be of any parity. So the parities of c_1 and d_1 already match. If the parity of d does not match that of c then we can replace d and d_1 using (6.22). This will change the parity of d because $p_1 - q_1$ is odd, but will leave the parity of d_1 unchanged because $p-q$ is even.

Summarizing, we can always make sure that c, d have the same parity, and so do c_1, d_1. The solutions a, b, a_1, b_1 are therefore integers, and $\mathbf{P}$ and $\mathbf{P}_1$ are coprime indeed. □

6.2.2 Embedding circulants into unimodular matrices

One outcome from the proof of Theorem is that when $\mathbf{P}$ and $\mathbf{P}_1$ are circulant and coprime, the matrices $\mathbf{A}$ and $\mathbf{A}_1$ in

$$\mathbf{P}\mathbf{A} + \mathbf{P}_1\mathbf{A}_1 = \mathbf{I} \tag{6.23}$$

can also be assumed to be circulant (see Eq. (6.8)). Since circulants commute we also have

$$\mathbf{A}\mathbf{P} + \mathbf{A}_1\mathbf{P}_1 = \mathbf{I} \tag{6.24}$$

The following matrix identity therefore follows:

$$\underbrace{\begin{bmatrix} \mathbf{P} & \mathbf{P}_1 \\ \mathbf{A}_1 & -\mathbf{A} \end{bmatrix}}_{\mathbf{U}} \underbrace{\begin{bmatrix} \mathbf{A} & \mathbf{P}_1 \\ \mathbf{A}_1 & -\mathbf{P} \end{bmatrix}}_{\mathbf{V}} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \tag{6.25}$$

where we have used the facts that $\mathbf{P}\mathbf{P}_1 - \mathbf{P}_1\mathbf{P} = \mathbf{0}$ and $\mathbf{A}\mathbf{A}_1 - \mathbf{A}_1\mathbf{A} = \mathbf{0}$. Taking determinants, we see from (6.25) that $\det\mathbf{V} = \det\mathbf{U} = \pm 1$. This shows, in particular, that when two circulants $\mathbf{P}$ and $\mathbf{P}_1$ are coprime and therefore satisfy (6.23), the integer matrix $\mathbf{U}$ indicated above is unimodular, and its inverse is the unimodular matrix $\mathbf{V}$. Note that all the indicated submatrices of $\mathbf{U}$ are circulants. So we say that when circulant matrices $\mathbf{P}$ and $\mathbf{P}_1$ are coprime, they can be *embedded* into a unimodular matrix $\mathbf{U}$ with circulant blocks.

6.2.3 Adjugate pairs

Given an arbitrary matrix $\mathbf{P}$, suppose $\mathbf{P}_1$ is related to it as follows:

$$\mathbf{P} = \begin{bmatrix} p & q \\ r & s \end{bmatrix}, \quad \mathbf{P}_1 = \begin{bmatrix} s & -q \\ -r & p \end{bmatrix} \tag{6.26}$$

Then we say that $\mathbf{P}_1$ is the adjugate of $\mathbf{P}$ (see [91] for the general definition). Notice that $\mathbf{P}\mathbf{P}_1 = \mathbf{P}_1\mathbf{P} = (ps - qr)\mathbf{I}$, so the matrices *automatically commute*. The importance of adjugate pairs of coprime matrices in 2D array processing is explained in Sec.V of [185]. For the special case where the two circulants are *adjugates* of each other, that is,

$$\mathbf{P} = \begin{bmatrix} p & q \\ q & p \end{bmatrix}, \quad \mathbf{P}_1 = \begin{bmatrix} p & -q \\ -q & p \end{bmatrix}, \tag{6.27}$$

we have $p_1 = p$ and $q_1 = -q$. In this case the coprimality conditions (6.9) reduce to the single condition

$$(p+q,\ p-q) = 1. \tag{6.28}$$

Eq. (6.28) says that *the two DFT coefficients of the sequence $\{p, q\}$ are coprime.* We claim that this is equivalent to the following condition:

$$(p, q) = 1, \quad \textit{and} \quad p \text{ and } q \text{ have opposite parity.} \tag{6.29}$$

Here "parity" refers to "odd" or "even" parity. "Opposite parity" means one integer is even and the other is odd.

Proof of equivalence. First assume (6.28) holds. If p and q both had a factor $g > 1$, it would be a factor of $p + q$ and $p - q$, violating (6.28). If p and q had the same parity then $p + q$ and $p - q$ would have a common factor of 2, again violating (6.28). Thus condition (6.28) certainly implies both conditions in (6.29). Conversely, suppose (6.29) holds. If $g > 1$ is the gcd of $p + q$ and $p - q$ then $p + q = ga, p - q = gb$ so that $2p = g(a + b)$ and $2q = g(a - b)$. But since $(p, q) = 1$ this is not possible unless $g = 2$ in which case $p+q$ and $p-q$ are even, violating the opposite-parity condition in (6.29). □

6.3 Skew Circulants

A skew-circulant or anticirculant matrix is similar to a circulant except for the extra negative signs below the main diagonal. Coprime pairs of skew circulant matrices have applications in two dimensional sampling (Sec. V. of [185]). In this section we consider pairs of skew-circulant matrices:

$$\mathbf{P} = \begin{bmatrix} p & q \\ -q & p \end{bmatrix}, \quad \mathbf{P}_1 = \begin{bmatrix} p_1 & q_1 \\ -q_1 & p_1 \end{bmatrix} \tag{6.30}$$

Direct multiplication shows that skew circulant matrices are commutative, and that the product is skew-circulant. We now derive the conditions under which the skew-circulants are coprime.

For the case of circulants, the derivation was facilitated by the fact that the vectors $[\,1 \quad 1\,]$ and $[\,1 \quad -1\,]$ are left-eigenvectors of circulants. This was implicitly involved in the proof of Theorem (see first paragraph of the proof). For a skew-circulant unfortunately, the eigenvectors are complex, and have the form $[\,1 \quad j\,]$ and $[\,1 \quad -j\,]$. A different technique will be used for the derivation of coprimality conditions, as elaborated next.

6.3.1 Derivation of the conditions for coprimality

From Bezout's theorem we know that coprimality holds if and only if there exist integers $a, b, c, d, a_1, b_1, c_1, d_1$ such that

$$\begin{bmatrix} p & q \\ -q & p \end{bmatrix} \underbrace{\begin{bmatrix} a & c \\ -b & d \end{bmatrix}}_{\mathbf{A}} + \begin{bmatrix} p_1 & q_1 \\ -q_1 & p_1 \end{bmatrix} \underbrace{\begin{bmatrix} a_1 & c_1 \\ -b_1 & d_1 \end{bmatrix}}_{\mathbf{A}_1} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{6.31}$$

Extracting the first column of this equation, we get

$$\begin{bmatrix} p & q \\ -q & p \end{bmatrix} \begin{bmatrix} a \\ -b \end{bmatrix} + \begin{bmatrix} p_1 & q_1 \\ -q_1 & p_1 \end{bmatrix} \begin{bmatrix} a_1 \\ -b_1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{6.32}$$

In view of the skew-circulant property of the matrices, this can be rearranged as

$$\begin{bmatrix} p & q \\ -q & p \end{bmatrix} \begin{bmatrix} b \\ a \end{bmatrix} + \begin{bmatrix} p_1 & q_1 \\ -q_1 & p_1 \end{bmatrix} \begin{bmatrix} b_1 \\ a_1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{6.33}$$

Comparing with the 2nd column of Eq. (6.31), we conclude that if Eq. (6.31) can at all be satisfied, then it can in particular be satisfied by restricting $\mathbf{A}$ and $\mathbf{B}$ to be skew-circulants:

$$\begin{bmatrix} p & q \\ -q & p \end{bmatrix} \begin{bmatrix} a & b \\ -b & a \end{bmatrix} + \begin{bmatrix} p_1 & q_1 \\ -q_1 & p_1 \end{bmatrix} \begin{bmatrix} a_1 & b_1 \\ -b_1 & a_1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{6.34}$$

Thus, $\mathbf{P}$ and $\mathbf{P}_1$ are coprime if and only if there exist integers a, b, a_1, b_1 satisfying Eq. (6.32), i.e., satisfying the following two equations:

$$pa - qb + p_1 a_1 - q_1 b_1 = 1 \tag{6.35}$$

$$qa + pb + q_1 a_1 + p_1 b_1 = 0 \tag{6.36}$$

By Euclid's theorem, Eq. (6.35) can be satisfied if and only if p, q, p_1, and q_1 have no common factor, i.e.,

$$(p, q, p_1, q_1) = 1. \tag{6.37}$$

When this necessary condition is satisfied, we can find the integers a, b, a_1, b_1 satisfying (6.35) using Euclid's algorithm [101], [127]. But since the solutions given by Euclid's algorithm are not unique, we can exploit this to modify a, b, a_1, b_1 such that the second equation (6.36) is *also* satisfied. Thus, given one solution a, b, a_1, b_1 for (6.35), modify it as follows:

$$\begin{aligned} &p(a + \alpha q + \beta p_1 + \gamma q_1) - q(b + \alpha p + \delta p_1 + \epsilon q_1) \\ &+ p_1(a_1 - \beta p + \delta q + \eta q_1) - q_1(b_1 + \gamma p - \epsilon q + \eta p_1) = 1 \end{aligned}$$

This equation continues to be satisfied no matter how the six integers $\alpha, \beta, \gamma, \delta, \epsilon$, and η are chosen (because they simply cancel off on the left hand side). Under certain conditions on p, q, p_1, and q_1, these free integers can be chosen such that (6.36) also holds, as we shall elaborate next. Substituting the modified values of a, b, a_1, and b_1 into (6.36), this equation becomes

$$\begin{aligned} &q(a + \alpha q + \beta p_1 + \gamma q_1) + p(b + \alpha p + \delta p_1 + \epsilon q_1) \\ &+ q_1(a_1 - \beta p + \delta q + \eta q_1) + p_1(b_1 + \gamma p - \epsilon q + \eta p_1) = 0, \end{aligned}$$

which can be rearranged as

$$\begin{aligned} &C + \alpha(p^2 + q^2) + (\beta - \epsilon)(p_1 q - q_1 p) \\ &+ (\gamma + \delta)(pp_1 + qq_1) + \eta(p_1^2 + q_1^2) = 0 \end{aligned} \tag{6.38}$$

Here $C \;\; qa + pb + q_1 a_1 + p_1 b_1$ is a constant for fixed p, q, p_1, q_1 and fixed initial solution a, b, a_1, b_1. If

$$\Big(p^2 + q^2, \;\; p_1^2 + q_1^2, \;\; p_1 q - q_1 p, \;\; pp_1 + qq_1\Big) = 1, \tag{6.39}$$

that is, the four integers on the left have gcd=1, then a linear combination of these four numbers (obtained through choice of $\alpha, \beta, \gamma, \delta, \epsilon, \eta$ which are completely free) can achieve any value, in particular, $-C$. So Eq. (6.38) can be satisfied indeed. Note that if $(p, q, p_1, q_1) = g > 1$ then the factor g (in fact, g^2) would be present on the left side of (6.39), and this equation cannot be satisfied. So, Eq. (6.39) also implies (6.37). In short, Eq. (6.39) is a sufficient condition for the existence of integers a, b, a_1, b_1 such that (6.34) holds.

Necessity of the sufficient condition. We have just shown that Eq. (6.39) is sufficient for coprimality of $\mathbf{P}$ and $\mathbf{P}_1$. We now show that Eq. (6.39) is actually also necessary! For this assume the contrary, and let $g > 1$ denote the gcd of the four integers on the left hand side of (6.39). Then we also have, in particular,

$$\Big(p^2 + q^2, \;\; p_1 q - q_1 p, \;\; pp_1 + qq_1\Big) \geq g > 1, \tag{6.40}$$

$$\Big(p_1^2 + q_1^2, \; p_1 q - q_1 p, \; pp_1 + qq_1\Big) \geq g > 1. \tag{6.41}$$

Assuming that Eq. (6.32) is still satisfied, we will bring about a contradiction. Premultiplying (6.32) by $[\, q_1 \quad p_1 \,]$ we get

$$[\, q_1 p - p_1 q \quad p p_1 + q q_1 \,]\, \mathbf{u} + [\, 0 \quad p_1^2 + q_1^2 \,]\, \mathbf{v} = q_1, \tag{6.42}$$

where $\mathbf{u}$ and $\mathbf{v}$ are integer vectors. Similarly premultiplying (6.32) by $[\, p_1 \quad -q_1 \,]$ yields

$$[\, p p_1 + q q_1 \quad p_1 q - q_1 p \,]\, \mathbf{u} + [\, p_1^2 + q_1^2 \quad 0 \,]\, \mathbf{v} = p_1. \tag{6.43}$$

Using (6.41) in the above equations, we see that the left hand sides have a factor g which shows that p_1 and q_1 on the right hand sides have a common factor g. Similarly by premultiplying (6.32) by $[\, q \quad p \,]$ and $[\, p \quad -q \,]$, and using (6.40) this time, we can verify that p and q have a common factor g. In short, we have shown that $(p, q, p_1, q_1) \geq g > 1$ which contradicts the necessary condition (6.37) for coprimality. This proves that, for coprimality of $\mathbf{P}$ and $\mathbf{P}_1$ it is necessary to have $g = 1$. Summarizing, we have proved that the skew-circulants in (6.30) are coprime if and only if

$$\Big(p^2 + q^2,\ \ p_1^2 + q_1^2,\ \ p_1 q - q_1 p,\ \ p p_1 + q q_1\Big) = 1. \tag{6.44}$$

We will show that the coprimality of the four integers in Eq. (6.44) is equivalent to the coprimality of the first three integers. In fact we can summarize the main result as follows:

Theorem . The skew-circulant matrices

$$\mathbf{P} = \begin{bmatrix} p & q \\ -q & p \end{bmatrix} \quad \text{and} \quad \mathbf{P}_1 = \begin{bmatrix} p_1 & q_1 \\ -q_1 & p_1 \end{bmatrix} \tag{6.45}$$

always commute, and they are coprime if and only if

$$\Big(p^2 + q^2,\ \ p_1^2 + q_1^2,\ \ p_1 q - q_1 p\Big) = 1, \tag{6.46}$$

or equivalently

$$\Big(p^2 + q^2,\ \ p_1^2 + q_1^2,\ \ p p_1 + q q_1\Big) = 1. \tag{6.47}$$

$\diamond$

Proof. Eq. (6.44) is clearly equivalent to

$$\Big(p^2+q^2,\ \ p_1^2+q_1^2,\ \ (p_1q-q_1p)^2,\ \ (pp_1+qq_1)^2\Big)=1. \tag{6.48}$$

Now use Fact from Sec. 6.1: adding the third entry to the fourth entry, and then dropping the square in the third entry, (6.48) becomes

$$\Big(p^2+q^2,\ \ p_1^2+q_1^2,\ \ p_1q-q_1p,\ \ (p^2+q^2)(p_1^2+q_1^2)\Big)=1.$$

Since the last entry above is a multiple of the first entry, it can be dropped. This results in (6.46). Eq. (6.47) is proved similarly, by replacing the third entry in (6.48) with the sum of the third and fourth entries. □

The following discussions provide further insights into the conditions given by Theorem .

6.3.2 Coprimality of determinants

Notice that p^2+q^2 and $p_1^2+q_1^2$ are the determinants of the two matrices in (6.45). If these determinants are coprime, then the matrices are evidently also coprime.[1] The preceding theorem asserts that coprimality of determinants, while sufficient, is not *necessary*, since (6.44) is weaker than coprimality of p^2+q^2 and $p_1^2+q_1^2$. For example, suppose $p=3, q=4, p_1=7$, and $q_1=9$. Then $p^2+q^2=25$ and $p_1^2+q_1^2=130$, so the determinants are not coprime. But $p_1q-q_1p=1$, and $pp_1+qq_1=57$, so Eq. (6.44) is still satisfied, and the skew circulants are coprime.

6.3.3 Adjugate pairs

Consider the special case where the skew circulants are adjugates of each other:

$$\mathbf{P}=\begin{bmatrix} p & q \\ -q & p \end{bmatrix} \quad \text{and} \quad \mathbf{P}_1=\begin{bmatrix} p & -q \\ q & p \end{bmatrix} \tag{6.49}$$

We claim that these matrices are coprime if and only if

$$(p,q)=1, \quad \text{and} \quad p \text{ and } q \text{ have opposite parity.} \tag{6.50}$$

[1] To see this, assume for example, that $\mathbf{L}$ is a glcd so that $\mathbf{P}=\mathbf{LX}$ and $\mathbf{P}_1=\mathbf{LY}$ for integer matrices $\mathbf{X}$ and $\mathbf{Y}$. Thus $\det \mathbf{P}$ and $\det \mathbf{P}_1$ have the common factor $\det \mathbf{L}$. But since these determinants are coprime, $\det \mathbf{L}=1$, that is $\mathbf{L}$ is unimodular, showing that $\mathbf{P}$ and $\mathbf{P}_1$ are left coprime.

This is identical to the coprimality condition (6.29) for an adjugate pair of circulants.

Proof of the claim. Eq. (6.45) corresponds to $p_1 = p, q_1 = -q$, so (6.44) becomes

$$\left(p^2 + q^2,\ \ 2pq,\ \ p^2 - q^2\right) = 1. \tag{6.51}$$

We will show that (6.50) is equivalent to (6.51). First assume (6.50) is not true. This could arise in one of two ways: if $(p, q) = g > 1$, then the factor g is there in $p^2 + q^2$, $2pq$, and $p^2 - q^2$, contradicting (6.51). If, on the other hand, p and q have the same parity, then $p^2 + q^2$, $2pq$, and $p^2 - q^2$, have the common factor 2, again contradicting (6.51). Thus if (6.50) is not true, then neither is (6.51). Next assume (6.50) is true. If (6.51) is not true, then, in particular, $(2pq, p^2 - q^2) = g$ for some $g > 1$. But the parity statement in (6.50) implies that $p^2 - q^2$ is odd, so g has to be a common factor of pq and $(p^2 - q^2)$. So there is a common factor $f > 1$ between p (say), and $p^2 - q^2$, hence between p^2 and $p^2 - q^2$. Subtracting, we find that f^2 is also a factor of q^2, so $(p^2, q^2) \geq f^2 > 1$, that is, $(p, q) > 1$, violating (6.50). This shows that (6.50) implies (6.51). □

6.3.4 Comparing with circulants

We know from Theorem that the two circulant matrices (6.5) are coprime if and only if Eq. (6.9) holds. For the special case where $p_1 = p$ and $q_1 = -q$ (adjugate pairs) this condition reduces to (6.29) which is identical to (6.50). That is, in the case of adjugate pairs, the condition for coprimality for circulants is the same as that for skew circulants. This raises the following question: *is the coprimality condition (6.9) for two general circulants equivalent to the coprimality condition (6.44) for two general skew circulants?* This turns out to be not the case. The following examples show that neither condition implies the other.

Example . Coprime skew-circulants. Suppose $p = 4, q = 1, p_1 = 5$, and $q_1 = 2$. Then $p^2 + q^2 = 17$, and $p_1^2 + q_1^2 = 29$, which already shows that (6.44) is satisfied. But $p - q = p_1 - q_1 = 3$, so (6.9) is violated. Thus the skew circulants (6.45) are coprime but the circulants (6.5) are not.

Example . Coprime circulants. Let $p = 4, q = 3, p_1 = 3$, and $q_1 = -4$. Then $p + q = 7, p_1 + q_1 = -1$, $p - q = 1$, and $p_1 - q_1 = 7$, so (6.9) is satisfied. But $p^2 + q^2 = p_1^2 + q_1^2 = p_1 q - q_1 p = 25$, and $pp_1 + qq_1 = 0$. Thus the four quantities on the left hand side of (6.44) have the factor 25, and (6.44) cannot be satisfied even though the necessary condition (6.37) is satisfied. As a double check, substituting the numbers into (6.42), which ought to be satisfied in the coprime case, we get $25X = -4$ which

cannot be satisfied for integer X. In this example, the circulants (6.5) are coprime but the skew circulants (6.45) are not.

Example . Coprime circulants and skew-circulants. For the example where $p = 3, q = 4, p_1 = 7$, and $q_1 = 9$, we found earlier that the skew circulants are coprime (though their determinants were not). Since $p + q = 7, p_1 + q_1 = 16, p - q = -1$, and $p_1 - q_1 = -2$, the circulants are coprime as well.

6.4 Triangular matrices

Next consider two triangular integer matrices of the form

$$\mathbf{P} = \begin{bmatrix} p & q \\ 0 & s \end{bmatrix}, \quad \mathbf{P}_1 = \begin{bmatrix} p_1 & q_1 \\ 0 & s_1 \end{bmatrix} \tag{6.52}$$

We present in this section the conditions for coprimality of these matrices. Even though these matrices do not commute as such, the following result can be considered to be important for its own sake. Furthermore, commutativity can be imposed separately in a number of ways, for example, by constraining the triangular matrices to be adjugate pairs (Sec. 6.4.1). Moreover, as shown in Sec. 6.5.3, an arbitrary pair of 2×2 integer matrices can be checked for coprimality by triangularizing these matrices using unimodular matrices, and then applying the theorem given next.

Theorem . The triangular matrices $\mathbf{P}$ and $\mathbf{P}_1$ are left-coprime if and only if

$$(s,\ s_1) = 1 \quad \text{and} \quad (p,\ p_1,\ q_1 s - q s_1) = 1, \tag{6.53}$$

i.e., if and only if s is coprime to s_1, and there is no factor common to p, p_1, and $q_1 s - q s_1$. $\diamond$

Proof. From Bezout's theorem we know that $\mathbf{P}$ and $\mathbf{P}_1$ are left-coprime if and only if there exist integers a, b, c, d and a_1, b_1, c_1, d_1 such that

$$\begin{bmatrix} p & q \\ 0 & s \end{bmatrix} \begin{bmatrix} a & b \\ c & d \end{bmatrix} + \begin{bmatrix} p_1 & q_1 \\ 0 & s_1 \end{bmatrix} \begin{bmatrix} a_1 & b_1 \\ c_1 & d_1 \end{bmatrix} = \begin{bmatrix} 1 & \times \\ 0 & 1 \end{bmatrix} \tag{6.54}$$

There is no loss of generality in taking the right hand side to be as above, and the entry denoted as $\times$ does not matter. This is because, the right hand side is unimodular, and its inverse is an integer matrix, so we can move it to the left side to make the preceding equation look like Bezout's identity.

The preceding equation is equivalent to the three scalar equations

$$pa + qc + p_1 a_1 + q_1 c_1 = 1 \tag{6.55}$$

$$sc + s_1 c_1 = 0 \tag{6.56}$$

$$sd + s_1 d_1 = 1 \tag{6.57}$$

Note that b and b_1 are not involved, and can be set to zero. The last equation shows that s and s_1 have to be coprime, in which case we can find d and d_1 to satisfy that equation. The second equation says $c/c_1 = -s_1/s$; since s_1/s is irreducible (from the third equation) this implies $c = -ns_1$ and $c_1 = ns$ for some integer n. Substituting this into the first equation results in $pa - qs_1 n + p_1 a_1 + q_1 sn = 1$ which can be rearranged as

$$pa + p_1 a_1 + (q_1 s - q s_1)n = 1. \tag{6.58}$$

The free integers a, a_1, and n can be chosen to satisfy this if and only if p, p_1, and $q_1 s - q s_1$ have no factor in common [101]. This completes the proof. □

6.4.1 Remarks

6.4.1.1 Adjugate pairs

For the special case where $\mathbf{P}_1$ is the adjugate of $\mathbf{P}$, that is,

$$\mathbf{P} = \begin{bmatrix} p & q \\ 0 & s \end{bmatrix}, \quad \mathbf{P}_1 = \begin{bmatrix} s & -q \\ 0 & p \end{bmatrix} \tag{6.59}$$

we have $p_1 = s, s_1 = p$, and $q_1 = -q$. Using (6.53) we see that the adjugate pair of triangular matrices (6.59) is coprime if and only if the single requirement

$$(p, s) = 1 \tag{6.60}$$

is satisifed.That is, the diagonal elements have to be coprime, and the value of the nonzero non diagonal element does not matter.

6.4.1.2 Right coprimality

The theorem provides conditions for *left coprimality*. It can similarly be shown that the two triangular matrices are *right coprime* if and only if

$$(p,\ p_1) = 1 \quad \text{and} \quad (s,\ s_1,\ q_1 p - q p_1) = 1. \tag{6.61}$$

Comparing (6.61) with (6.53) we find that the two triangular matrices are *left and right coprime* if and only if

$$(s,\ s_1) = 1 \quad \text{and} \quad (p,\ p_1) = 1. \tag{6.62}$$

This is because the second conditions in Eqs. (6.61) and (6.53) are automatically satisfied if (6.62) is satisfied. Eq. (6.62) says that the corresponding diagonal elements of the matrices must be coprime. The non diagonal elements do not matter here.

6.4.1.3 Commuting triangles

It is easily verified that the triangular matrices (6.52) commute if and only if

$$q_1 p - q p_1 = q_1 s - q s_1. \tag{6.63}$$

In this case the condition for left coprimality (6.53) and right coprimality (6.61) indeed become identical as expected.

6.5 Other related matrices

If $\mathbf{P}$ and $\mathbf{P}_1$ are left coprime, then $\mathbf{PU}$ and $\mathbf{P}_1\mathbf{V}$ are also left coprime for any pair of unimodular matrices $\mathbf{U}$ and $\mathbf{V}$. This is because the Bezout identity $\mathbf{PA} + \mathbf{P}_1\mathbf{B} = \mathbf{I}$ can be rewritten as $\mathbf{PUU}^{-1}\mathbf{A} + \mathbf{P}_1\mathbf{VV}^{-1}\mathbf{B} = \mathbf{I}$, where $\mathbf{U}^{-1}\mathbf{A}$ and $\mathbf{V}^{-1}\mathbf{B}$ continue to be integer matrices (since $\mathbf{U}$ and $\mathbf{V}$ are unimodular), hence Bezout's identity is satisfied by $\mathbf{PU}$ and $\mathbf{P}_1\mathbf{V}$. Using this simple idea, a number of related results for coprime pairs can be derived, starting from the results given in the earlier sections.

6.5.1 Reflected triangles

As a first example, we can generate a so-called *reflected triangle* from a triangle by post multiplying with a unimodular matrix:

$$\begin{bmatrix} p & q \\ 0 & s \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} q & p \\ s & 0 \end{bmatrix} \tag{6.64}$$

The matrix on the right is called a reflected triangle because it is the reflection of a triangular matrix with respect to a vertical (or horizontal) line in the middle. Since post multiplication by a unimodular matrix does not affect left-coprimality, we use Theorem to conclude that the reflected triangular matrices

$$\begin{bmatrix} q & p \\ s & 0 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} q_1 & p_1 \\ s_1 & 0 \end{bmatrix} \tag{6.65}$$

are *left-coprime* if and only if

$$(s,\ s_1) = 1 \quad \text{and} \quad (p,\ p_1,\ q_1 s - q s_1) = 1. \tag{6.66}$$

Similarly, since

$$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} p & q \\ 0 & s \end{bmatrix} = \begin{bmatrix} 0 & s \\ p & q \end{bmatrix} \tag{6.67}$$

we see that the reflected triangles

$$\begin{bmatrix} 0 & s \\ p & q \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} 0 & s_1 \\ p_1 & q_1 \end{bmatrix} \tag{6.68}$$

are *right coprime* if and only if

$$(p,\ p_1) = 1 \ \text{ and } \ (s,\ s_1,\ q_1 p - q p_1) = 1. \tag{6.69}$$

Conditions for right coprimality of (6.65) and left coprimality of (6.68) can similarly be derived.

6.5.2 General matrices with three freedoms

Circulants and skew circulants are characterized by two distinct integers. Triangular matrices and their minor variations (e.g., antisymmetric) are characterized by three distinct integers. For another variation, consider the following unimodular postmultiplication of a triangular matrix:

$$\begin{bmatrix} p & q \\ 0 & s \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} p+q & q \\ s & s \end{bmatrix} \tag{6.70}$$

This is a matrix with a repeated element s, so three integers determine the entire matrix. For pairs of matrices of this kind, the conditions for left coprimality can be readily derived from Theorem . Other examples of matrices with three distinct elements can be generated by appropriately modifying the unimodular matrix in Eq. (6.70).

6.5.3 Testing coprimality by triangularization

We conclude by pointing out another important consequence of the fact that (left) coprimality is preserved when unimodular matrices are multiplied on the right. For this we first show that any 2×2 integer matrix $\mathbf{P}$ can be *triangularized* by multiplication with a unimodular matrix. That is, we can choose the elements u_k of a unimodular matrix $\mathbf{U}$ such that

$$\underbrace{\begin{bmatrix} p & q \\ r & s \end{bmatrix}}_{\mathbf{P}} \underbrace{\begin{bmatrix} u_1 & u_2 \\ u_3 & u_4 \end{bmatrix}}_{\mathbf{U}} = \underbrace{\begin{bmatrix} a & b \\ 0 & d \end{bmatrix}}_{\mathbf{\Delta}_p} \tag{6.71}$$

for some integers a, b, and d. For this it is first necessary to force $ru_1 + su_3 = 0$. Notice that unimodularity implies $u_1u_4 - u_2u_3 = 1$, so the integers u_1 and u_3 are required to be coprime. To ensure this we proceed as follows: with g denoting the gcd of r and s, write

$$r = gX, \quad s = gY \tag{6.72}$$

so that X and Y are coprime. Choosing $u_1 = Y$ and $u_3 = -X$ we ensure then that they are coprime, and furthermore

$$ru_1 + su_3 = g(u_1X + u_3Y) = g(YX - XY) = 0 \tag{6.73}$$

indeed. Since u_1 and u_3 are coprime by construction, there exists (by Euclid's theorem) u_2 and u_4 such that $u_1u_4 - u_2u_3 = 1$, that is, $\mathbf{U}$ is unimodular indeed! Notice that (6.71) is analogous (though by no means equivalent) to the *QR decomposition* for complex matrices where $\mathbf{Q}$ is unitary rather than unimodular, and $\mathbf{R}$ is triangular.

The importance of the preceding triangualrization is the following: given two integer matrices $\mathbf{P}$ and $\mathbf{Q}$, suppose we triangularize them both:

$$\mathbf{PU} = \mathbf{\Delta}_p, \quad \mathbf{QV} = \mathbf{\Delta}_q. \tag{6.74}$$

Since $\mathbf{U}$ and $\mathbf{V}$ are unimodular, it follows that $\mathbf{P}$ and $\mathbf{Q}$ are left coprime if and only if $\mathbf{\Delta}_p$ and $\mathbf{\Delta}_q$ are left coprime. And since the latter can be checked readily using Theorem , coprimality of the arbitrary pair $\{\mathbf{P}, \mathbf{Q}\}$ is readily checked by this triangularization trick! There is no need to compute Smith forms and so forth, as in more elaborate test procedures.

6.5.4 Unimodular similarity transforms

While we are on the topic of unimodular matrices, we mention an important transformation here which preserves coprimeness and commutativity [72]. Given a commuting coprime pair of matrices $\mathbf{P}, \mathbf{Q}$, define a new pair

$$\mathbf{P}_1 = \mathbf{UPU}^{-1}, \quad \mathbf{Q}_1 = \mathbf{UQU}^{-1} \tag{6.75}$$

where $\mathbf{U}$ is unimodular (so that $\mathbf{U}^{-1}$ is an integer matrix). The commutativity property $\mathbf{P}_1\mathbf{Q}_1 = \mathbf{Q}_1\mathbf{P}_1$ is readily verified. Since coprimality of $\mathbf{P}$ and $\mathbf{Q}$ implies $\mathbf{PA} + \mathbf{QB} = \mathbf{I}$, it follows that $\mathbf{UPU}^{-1}\mathbf{UA} + \mathbf{UQU}^{-1}\mathbf{UB} = \mathbf{U}$, that is,

$$\underbrace{\mathbf{UPU}^{-1}}_{\mathbf{P}_1}\underbrace{\mathbf{UAU}^{-1}}_{\mathbf{A}_1} + \underbrace{\mathbf{UQU}^{-1}}_{\mathbf{Q}_1}\underbrace{\mathbf{UBU}^{-1}}_{\mathbf{B}_1} = \mathbf{I}. \tag{6.76}$$

Since $\mathbf{A}_1$ and $\mathbf{B}_1$ are integer matrices it follows that $\mathbf{P}_1$ and $\mathbf{Q}_1$ are also coprime. *Summarizing, if* $\mathbf{P}$ *and* $\mathbf{Q}$ *are commuting and coprime, then so are* $\mathbf{P}_1$ *and* $\mathbf{Q}_1$. That is, a unimodular similarity transformation preserves the commuting coprime property.[2] For example, suppose $\mathbf{P}$ and $\mathbf{Q}$ are circulants,

[2] A transform of the type $\mathbf{T}^{-1}\mathbf{AT}$ is called a similarity transform. It preserves the eigenvalues, hence the trace and the determinant.

and we apply a unimodular similarity transform as follows:

$$\underbrace{\begin{bmatrix} 1 & 0 \\ n & 1 \end{bmatrix}}_{\mathbf{U}} \underbrace{\begin{bmatrix} p & q \\ q & p \end{bmatrix}}_{\mathbf{P}} \underbrace{\begin{bmatrix} 1 & 0 \\ -n & 1 \end{bmatrix}}_{\mathbf{U}^{-1}} = \underbrace{\begin{bmatrix} p-nq & q \\ q-n^2q & p+nq \end{bmatrix}}_{\mathbf{P}_1} \tag{6.77}$$

and similarly for $\mathbf{Q}$. Then $\mathbf{P}_1$ and $\mathbf{Q}_1$ are not circulant and appear to have no specific structure at all, yet they commute, and their coprimality can be tested readily using Theorem for circulants.

6.6 3×3 Triangular matrix and adjugate

All the cases considered so far are for 2×2 matrices. Although the coprimality conditions are difficult to derive for 3×3 matrices in general, there is one example which can be handled with the tools we have described so far. This is the case of a 3×3 triangular matrix and its adjugate:

$$\mathbf{P} = \begin{bmatrix} p & q & r \\ 0 & s & t \\ 0 & 0 & u \end{bmatrix}, \quad \widehat{\mathbf{P}} = \begin{bmatrix} su & -qu & qt-sr \\ 0 & pu & -pt \\ 0 & 0 & ps \end{bmatrix}. \tag{6.78}$$

We will prove the following result:

Theorem . The triangular matrix $\mathbf{P}$ and its adjugate given in Eq. (6.78) are left coprime if and only if

$$(p,s)=1,\ (p,u)=1, \quad \text{and} \quad (s,u)=1, \tag{6.79}$$

that is, the diagonal elements of $\mathbf{P}$ are coprime in pairs.$\diamond$

Notice that the result is sysmmetric in the diagonal elements p, s, u. Furthermore, the non diagonal elements of the triangular matrix do not have a role in coprimality.

Proof of Theorem . First assume (6.79) is true. Then the corresponding diagonal elements of $\mathbf{P}$ and $\widehat{\mathbf{P}}$ are coprime:

$$(p,su)=1,\ (s,pu)=1, \quad \text{and} \quad (u,ps)=1 \tag{6.80}$$

So there exist integers a_k, b_k such that $pa_1 + (su)b_1 = 1$, $sa_2 + (pu)b_2 = 1$, $ua_3 + (ps)b_3 = 1$, that is, there exist diagonal integer matrices $\mathbf{\Lambda}_a, \mathbf{\Lambda}_b$ such that

$$\mathbf{P}\mathbf{\Lambda}_a + \widehat{\mathbf{P}}\mathbf{\Lambda}_b = \begin{bmatrix} 1 & \times & \times \\ 0 & 1 & \times \\ 0 & 0 & 1 \end{bmatrix} \tag{6.81}$$

The right hand side $\mathbf{\Delta}$ is triangular because all matrices on the left are triangular. Since $\mathbf{\Delta}$ is also unimodular we can convert (6.81) to Bezout's identity by multiplying with $\mathbf{\Delta}^{-1}$. Thus, $\mathbf{P}$ and $\widehat{\mathbf{P}}$ are left coprime.

For the converse, we use a comment made at the end of Sec. 6.1.1 under "Integers on a row or column." If $\mathbf{P}$ and $\widehat{\mathbf{P}}$ are coprime, then in particular the elements on the last row of $[\,\mathbf{P} \quad \widehat{\mathbf{P}}\,]$ are coprime, that is, $(u, ps) = 1$. Thus $(u, p) = 1$ and $(u, s) = 1$ are necessary; it only remains to show that $(p, s) = 1$ is also necessary. For this, note that if $\mathbf{P}$ and its adjugate $\widehat{\mathbf{P}}$ are left coprime, then they are also *right* coprime (because $\mathbf{P}$ and $\widehat{\mathbf{P}}$ commute). So, elements along any *column* of the $2D \times D$ matrix $\begin{bmatrix} \mathbf{P} \\ \widehat{\mathbf{P}} \end{bmatrix}$ have gcd $= 1$. Since the first column of the above $2D \times D$ matrix only has p and su (besides 0s) we have $(p, su) = 1$, implying $(p, s) = 1$ indeed. □

Consider now the special case where the triangular matrix $\mathbf{P}$ is *also* Toeplitz. Then $\mathbf{P}$ and its adjugagte $\widehat{\mathbf{P}}$ have the form

$$\mathbf{P} = \begin{bmatrix} p & q & r \\ 0 & p & q \\ 0 & 0 & p \end{bmatrix}, \quad \widehat{\mathbf{P}} = \begin{bmatrix} p^2 & -pq & q^2 - pr \\ 0 & p^2 & -pq \\ 0 & 0 & p^2 \end{bmatrix} \tag{6.82}$$

Since the diagonal elements of $\mathbf{P}$ are identical ($p = s = u$), the condition (6.79) reduces to $(p, p) = 1$, i.e., $p = \pm 1$. Thus, the triangular Toeplitz matrix and its adjugate in (6.82) are coprime if and only if $\mathbf{P}$ has all diagonal elements equal to ± 1. In this case $\det \mathbf{P} = \pm 1$, that is, $\mathbf{P}$ is essentially unimodular.

6.7 Concluding remarks

In this chapter we first derived conditions under which certain classes of 2×2 matrix pairs are coprime. Two of these classes, namely circulants and skew-circulants, contain only commuting matrices. For each class, the special case where the pairs of matrices are adjugates of each other

was also considered, because such pairs automatically commute. In this way, we have developed several families of commuting coprime matrices. In each case that was considered, the coprimality of the matrices was reduced to the coprimality of two or more scalar integers related to those matrices.

We also showed that coprimality of arbitrary pairs of matrices can be tested by testing the coprimality of a pair of triangular matrices, which in turn can be done by using Theorem . The results on individual classes of coprime matrices continue to be valuable in spite of this result because, the individual theorems allow us to constructively generate entire families. For example, the "skew-circulant and adjugate pair" family, and similarly the "circulant and adjugate pair" family can be generated simply by generating a pair of coprime integers with opposite parity.

We concluded by considering a 3×3 triangular matrix $\mathbf{P}$ and its adjugate, which were shown to be coprime if and only if the three diagonal elements of $\mathbf{P}$ are pairwise coprime. Thus, by generating three integers which are pairwise coprime, this entirely family can be systematically constructed.

Chapter 7

A Correlation Aware Framework For Sparse Estimation: Ideal Covariance Matrix

7.1 Introduction and Related Work

The Sparse Sampler and Arrays and new subspace based reconstruction algorithms proposed in the earlier chapters demonstrated that it is possible to identify/estimate more parameters than number of physical measurements in a largely underdetermined linear measurement model. In this chapter, we will consider additional structure on the unknown signal by assuming that it is sparse, i.e., it has only a few non zero elements. This sparse linear measurement model occurs centrally in *compressive sensing* [24, 56] where the goal is to recover the sparse signal by computing the sparsest solution to an underdetermined set of linear equations. There is a rich history of theoretical and algorithmic developments for this problem, which have been outlined in Chapter 1.

In this chapter, we introduce a statistical model on the unknown signal and assume the *knowledge of priors* in the form of additional information about the nature of correlation between the non zero components of the unknown signal. We will propose a new approach to "sparse support recovery" based on the covariance matrix of the observations and demonstrate how this can fundamentally lead to the possibility of recovering supports of much larger size than most existing approaches. Many of the well known approaches to sparse reconstruction, such as convex formulation [35, 28], greedy algorithms [177, 180, 128], or non convex norm based formulations [77, 83, 94] treat the recovery problem rather deterministically in the sense that they do not exploit the statistical properties of the unknown signal. In few cases, the sparse signal recovery problem has been

cast in an estimation/detection setting [51]. More generally, a family of recovery techniques which explicitly compute a MAP-like estimate of the sparse unknown vector, have been unified under the umbrella of Bayesian Compressive Sensing [95]. Sparse Bayesian Learning [176, 199, 200, 207] provides an alternative to the usual l_1 minimization approach to recovery of such sparse vectors by computing the MAP estimate of the sparse unknown vector. Few other kinds of priors have also been considered in recent literature. They include (a) knowledge of feasible configurations of the non zero elements of the sparse vector in a model-based recovery technique [9], (b) partial knowledge of the support [188], (c) knowledge of the probability with which each index can be non zero in he context of a non uniform sparse model [99], and (d) knowledge of the rank of the sparse signal matrix (in the multiple measurement vector model) [53]. However, none of these frameworks explicitly uses the correlation between the non zero values of the sparse signal as a prior.

In contrast to existing approaches to sparse recovery which use priors of various kinds, our formulation specifically makes use of the following prior:

The non zero elements of the sparse vector are assumed statistically uncorrelated

This was a fundamental assumption in the preceding chapters for direction finding and spectral estimation. We will show the implication of this assumption with respect to the size of the support that can be recovered. In particular, our support recovery algorithm will act on the data covariance matrix instead of the raw measurements themselves. Such estimation techniques which use correlation of the signal find extensive use in a wide variety of parametric/non parametric estimation problems. In particular any spectral estimation algorithm fundamentally relies upon using the autocorrelation of the (WSS) signal. Such applications include recent approaches to spectral compressive sensing [66], wideband spectrum sensing for cognitive radio [153, 174, 203], direction finding in antenna arrays [121, 206, 166], radar [87], MEG/EEG imaging and source localization [201, 208].

7.2 Related Work

The body of work in sparse sensing and recovery that are most related to our work consists of sparse spectral estimation techniques [174, 153, 121, 206, 166], methods for joint support recovery [107], and the general framework of Sparse Bayesian Learning and its recent applications [176, 207, 208]. None of these methods however explicitly uses the fact that the non zero elements

of the unknown signal are uncorrelated, although some of them implicitly make that assumption in their signal model. The problem of spectrum sensing in cognitive radio is cast as a sparse recovery problem in [153, 174] but no special sampling or reconstruction scheme that can directly exploit the lack of correlation among the different frequency bands are used. Similarly in [121, 166], different approaches to sparse modeling and estimation in direction finding problems have been developed without explicitly using the aforementioned fact. The work of [107] uses a hybrid of subspace based methods like MUSIC and l_1 norm minimization algorithms for joint support recovery. Although the computations are based on the covariance matrix of the data, it does not exploit the uncorrelated structure of the unknown signal. Similarly, the SBL framework models the unknown vector as a Gaussian random vector with diagonal covariance matrix, but it does not investigate what advantage this structure might offer. The use of SBL in multiple measurement model where temporal correlation might be present, was demonstrated in [208]. This is also different from our framework where we solely concentrate on spatial correlation (or the lack of) in the measurements. More recently, the work in [206] represents the sparse recovery problem using the covariance matrix in a form that closely matches our own formulation; however, they do not focus on developing new sampling schemes and arrays which can exploit the correlation priors. The use of Kronecker product matrices in compressive sensing has been studied in [65]. We however consider a special matrix product, namely the Khatri-Rao product [100], which, inspite of being derivable from the Kronecker product, has very different properties from the Kronecker product and the results in [65] are not quite applicable to our case.

Summarizing, almost every existing sparse recovery technique, be deterministic or statistical, operates in a regime where the sparsity is less than the size of the measured/observed vector. Using a certain kind of correlation priors, as described earlier, we will show that it is possible to recover sparse support of size larger than that of the observed vector(s). The design of measurement matrix will be shown to play a crucial role in achieving this, whereby we will develop new conditions on the matrix for such recovery to be successful. We will next review some existing guarantees for sparse recovery in Sec. 7.3 and provide an outline of the major contributions in Sec. 7.4.

Parts of this chapter have been presented in [141].

7.3 Review of Conditions for Uniqueness of Sparse Representations

A central problem in the field of Compressive Sensing and Sparse Reconstruction is the recovery of the sparsest solution to a linear underdetermined systems of equations which is also known as the Single Measurement Vector (SMV) model. Such a model is given by

$$\mathbf{y} = \mathbf{A}\mathbf{x}_0 \qquad \text{(SMV)} \tag{7.1}$$

where $\mathbf{A} \in \mathbb{C}^{M \times N}$ is a fat matrix with $N \gg M$ and $\mathbf{x}_0$ is a sparse vector with $\mid \text{Supp}(\mathbf{x}_0) \mid \ll N$. The problem of recovering the sparsest vector satisfying (7.1) is of central interest in many applications pertaining to compressive sampling and sparse reconstruction. The problem is cast as

$$\begin{aligned} &\min_{\mathbf{x}} \|\mathbf{x}\|_0 \qquad (P0)_{SMV} \\ &\text{subject to } \mathbf{y} = \mathbf{A}\mathbf{x} \end{aligned} \tag{7.2}$$

The sole property of $\mathbf{A}$ that decides when such recovery is possible, is the Kruskal Rank [105] of $\mathbf{A}$, denoted as Krank$(\mathbf{A})$, as defined below:

Definition 7.1: (**Kruskal Rank:**) The Kruskal Rank of $\mathbf{A}$, denoted as Krank$(\mathbf{A})$ is defined as the largest integer r such that *every* r columns of $\mathbf{A}$ are linearly independent. □

Denoting S_0 as the support of $\mathbf{x}_0$, the following condition provides a maximum limit on $\mid S_0 \mid$ (also known as the *sparsity level*) that ensures unique recovery of $\mathbf{x}_0$ [58] :

Theorem 7.3.1. *Consider the (SMV) model* $\mathbf{y} = \mathbf{A}\mathbf{x}_0$ *with* S_0 *denoting the support of* $\mathbf{x_0}$*. Solving* $(P0)_{SMV}$ *will uniquely recover* $\mathbf{x}_0$ *if and only if*

$$\mid S_0 \mid \leq \frac{\mathit{Krank}(\mathbf{A})}{2} \tag{7.3}$$

□

This guarantee on the maximum level of sparsity that can be recovered is fundamental in the sense that it needs to be satisfied for *any* recovery algorithm (convex or otherwise) to be successful.

Since Krank($\mathbf{A}$) $\leq$ rank($\mathbf{A}$), it can be easily shown that the maximum level of recoverable sparsity from the SMV model satisfies

$$\mid S_0 \mid \leq \frac{M}{2} \tag{7.4}$$

A natural extension to the SMV model is the Multiple Measurement Vector (MMV) Model:

$$\mathbf{Y}_L = \mathbf{A}\underbrace{[\mathbf{x}_1 \quad \mathbf{x}_2 \cdots \mathbf{x}_L]}_{\mathbf{X}_L} \qquad \text{(MMV)} \tag{7.5}$$

Here a set of $L > 1$ unknown sparse vectors $\mathbf{x}_l \in \mathbb{C}^{N\times 1}, l = 1, 2, \cdots L$ share a common support S_0, i.e., $\text{Supp}(\mathbf{x}_l) = S_0, \forall l$. Let $\mathbf{r}_{\mathbf{X}_L} \in \mathbb{R}^{N\times 1}$ be such that

$$[\mathbf{r}_{\mathbf{X}_L}]_i = \left(\sum_{l=1}^{L}([\mathbf{X}_L]_{n,l})^q\right)^{\frac{1}{q}}, \text{ for some } q \geq 1$$

The corresponding problem for recovering the sparsest set of vectors now becomes

$$\min_{\mathbf{X}\in\mathbb{C}^{N\times L}} \|\mathbf{r}_{\mathbf{X}_L}\|_0 \qquad (P0)_{MMV} \tag{7.6}$$

$$\text{subject to } \mathbf{Y}_L = \mathbf{A}\mathbf{X} \tag{7.7}$$

The recovery guarantee (independent of any specific algorithm used) for $\mid S_0 \mid$ now obeys [36]

Theorem 7.3.2. *Consider the MMV model (7.5) where S_0 denotes the common support of the L vectors. Solving $(P0)_{MMV}$ will recover the support S_0 if and only if*

$$\mid S_0 \mid < \frac{\textit{Krank}(\mathbf{A}) + \textit{rank}(\mathbf{Y}_L)}{2} \tag{7.8}$$

□

Since rank($\mathbf{Y}_L$) $\leq M$ and Krank($\mathbf{A}$) $\leq M$, (7.8) implies

$$\mid S_0 \mid < M. \tag{7.9}$$

It is to be noted that the maximum recoverable sparsity is independent of L as long as $L > M$.

From Theorems 7.3.1 and 7.3.2 it is clear that with no further assumptions on $\mathbf{x}_0$ or $\mathbf{X}_L$, *the*

maximum sparsity level that can be recovered from SMV or MMV model is always $O(M)$. Almost every existing algorithm for sparse reconstruction obeys this upper limit. In particular, no existing algorithm for sparse reconstruction considers sparsity levels which can be in the region $\mid S_0 \mid > M$. In this chapter, we will explore precisely this regime of sparsity levels which can be *larger than* M and show that it will be possible to recover supports of size potentially larger than M, provided a statistical framework for the unknown $\mathbf{x}$ is used with appropriate priors on the distribution.

7.4 Our Contributions

We summarize our main contributions along with an outline for the remainder of the chapter.

7.4.1 A Correlation Aware Framework

We will develop a novel approach to support recovery (i.e. recovery of S_0) incorporating a statistical framework in the SMV model. In particular, rather than treating $\mathbf{x}_0$ as a deterministic unknown vector, we model it to be a sparse random vector with a fixed support. In particular we will

1. Explicitly use statistical priors on the unknown sparse vector by imposing a structure on the covariance matrix of $\mathbf{x}$.

2. Propose novel conditions on the measurement matrix (which, in particular implies new strategies for structured sampling) and new reconstruction algorithms that can exploit the aforementioned priors.

We will establish that such correlation awareness, coupled with suitable designs for measurement matrices and appropriate algorithm can *guarantee the recovery of sparsity levels which can be orders of magnitude higher* than existing methods which are blind to the correlation structure. Specifically, we make the following assumptions regarding the statistics of $\mathbf{x}_0$:

1. (A1): $E(\mathbf{x}_0) = \mathbf{0}$

2. (A2): $E([\mathbf{x_0}]_i[\mathbf{x_0}]_j^*) = 0, i \neq j, i, j \in S_0$

Assumption (A2) means that the non zero elements of $\mathbf{x}_0$ are statistically uncorrelated. In what follows, we will develop a new framework for recovering S_0 that exploits this fact. The main idea

is to use the data covariance matrix

$$\mathbf{R_{yy}} = E(\mathbf{y}\mathbf{y}^H) \tag{7.10}$$

for recovery of S_0. The first part of our results is based on the *exact* knowledge of the data covariance matrix $\mathbf{R_{yy}}$ whereby we will show it is possible to guarantee the recovery of sparsity levels as large as

$$| S_0 | = O(M^2). \tag{7.11}$$

This becomes possible under certain conditions on the measurement matrix. Sparse sampling schemes such as nested and coprime sampling will be shown to satisfy these conditions and hence they become natural choices for correlation aware support recovery. Without the knowledge of correlation priors and appropriate sampling strategies, existing performance bounds on sparse models [60, 59, 29, 189, 190, 172] can only guarantee recovery of $O(M)$ sparsity levels. In the second part, we will also consider the case when the matrix $\mathbf{R_{yy}}$ is not known exactly and we need to rely upon estimated $\hat{\mathbf{R}}_{\mathbf{yy}}$ (computed from finite number of measurements).

7.4.2 Recovery of Sparse Support v/s Sparse Vector

We want to highlight an important difference between the recovery of the sparse vector(s) $\mathbf{x}$ and the recovery of its support. In all existing approaches to sparse recovery, no distinction is made between the two. In fact, in every case, the recovery of support S_0 automatically leads to recovery of the sparse vector $\mathbf{x}$ since it is possible to solve $\mathbf{y} = \mathbf{A}_{S_0}[\mathbf{x}]_{S_0}$ uniquely for $[\mathbf{x}]_{S_0}$. This happens because $| S_0 | <$ Krank($\mathbf{A}$). In our correlation aware framework, we will demonstrate that it is *possible to recover support* S_0 where $| S_0 | \gg M$, possibly $| S_0 | = O(M^2)$. However, in this case, it is *not possible to recover the vector(s)* $\mathbf{x}$ *itself*. This can be easily seen from the fact that when $| S_0 | > M \geq$ rank($\mathbf{A}$), there is no unique $[\mathbf{x}_0]_{S_0}$ satisfying $\mathbf{y} = \mathbf{A}_{S_0}[\mathbf{x}]_{S_0}$. Our framework explicitly brings out this important difference between the two kinds of recovery which has not been considered in the literature so far. Also, as we will show later, it is possible to recover $E(|[\mathbf{x}]_i|^2), i \in S_0$ although it is not possible to recover $\mathbf{x}_{S_0}$ itself.

In many applications, the support S_0 itself is sought after rather than the sparse vector $\mathbf{x}$ itself. They include spectral estimation [153, 174], imaging [208, 201, 117], direction finding in array pro-

cessing [166, 206, 121, 107], multivariate regression and classification [129, 130] etc. In most of these applications, the information of interest is actually contained in the support. Such applications can potentially benefit greatly from our proposed framework which can fundamentally allow recovery of much larger supports.

The rest of the chapter is organized as follows. In Section 7.5, we will establish fundamental bounds on the size of the largest sparse support that can be recovered from *an exact knowledge of the data covariance matrix* $\mathbf{R_{yy}}$ and demonstrate that this can be as large as $O(M^2)$. We will characterize the role of Khatri Rao product of the measurement matrix $\mathbf{A}$ in ensuring this and how this leads to interesting and non trivial design problems. In Section 7.6, we will develop new conditions for convex relaxation of the support recovery problem using our proposed correlation aware framework. Some of the guarantees will be based on coherence of the measurement matrix and they strictly improve similar existing results based on coherence. In sec 7.7, we will also show that the support recovery problem becomes equivalent to minimizing the l_1 norm of a certain positive vector (derived from the covariance matrix) satisfying a set of underdetermined linear equations and the positive can be exploited to guarantee recovery of $O(M^2)$ sparsity levels even for the relaxed problem. We will also provide an illustrative application, namely direction finding in antenna arrays, which will bring out the role of antenna geometry in sparse support recovery and will unify the proposed nested and coprime sampling framework. Finally in Section 7.8, we will re-derive similar guarantees for greedy heuristics such as the Orthogonal Matching Pursuit (OMP) for the correlation aware framework and establish that correlation awareness strictly improves upon existing recovery guarantees in this case as well.

7.5 Uniqueness of Support Recovery From Exact Data Covariance

In this section, we will establish a set of new guarantees on the size of recoverable $\mid S_0 \mid$ which are fundamentally related to the uniqueness of sparse representation. These conditions will be for support recovery from the data covariance matrix, as opposed to from the measurements themselves. In sec. 7.6, we will develop convex relaxations for the support recovery problem under the assumption that we still know $\mathbf{R_{yy}}$ exactly. In Chapter 8, uniqueness conditions and guarantees for convex relaxations will be derived when only an estimate $\hat{\mathbf{R}}_{\mathbf{yy}}$ is available.

1. We know the ideal data covariance matrix $\mathbf{R_{yy}} = E(\mathbf{y}\mathbf{y}^{\mathbf{H}})$

2. Assumptions (A1) and (A2) hold.

The exact knowledge of $\mathbf{R_{yy}}$ is possible when large number of measurements are available, i.e. as $L \to \infty$ in the model (7.5). Notice that Theorem 7.3.1 dictates that without any assumptions on $\mathbf{x}_l$ other than sparsity, it is not possible to recovery sparsity levels higher than M regardless of how large L might be. We will show that by exploiting priors given by (A1) and (A2), we can actually overcome this limitation and recover sparsity level as large as $O(M^2)$ when we know $\mathbf{R_{yy}}$ exactly. This is possible as $L \to \infty$ since

$$\lim_{L\to\infty} \frac{1}{L}\mathbf{Y}_L\mathbf{Y}_L^H = \mathbf{R_{yy}} \tag{7.12}$$

Our goal is to first develop a set of conditions to investigate if it is at all possible to increase the recoverable sparsity limit to $O(M^2)$ when we have *exact knowledge* of $\mathbf{R_{yy}}$.

The following lemma establishes the connection of support recovery from the covariance matrix:

Lemma 7.5.1. *Consider a SMV model (7.1) where S_0 denotes the support of $\mathbf{x}_0$. Under assumptions (A1) and (A2), recovery of S_0 is equivalent to recovering* $\mathrm{Supp}(\mathbf{p})$ *in*

$$\mathbf{z} = \Big(\mathbf{A}^* \odot \mathbf{A}\Big)\mathbf{p} \qquad (Co - SMV) \tag{7.13}$$

where $\mathbf{z} = \mathrm{vec}(\mathbf{R_{yy}})$ *and* $\mathbf{p} = diag(\mathbf{R_{xx}})$. □

Proof. The covariance matrix of $\mathbf{y}$ is given by

$$\mathbf{R_{yy}} = \mathbf{A}\mathbf{R_{xx}}\mathbf{A}^H$$

Upon vectorization, we obtain

$$\mathbf{z} = (\mathbf{A}^* \otimes \mathbf{A})\mathrm{vec}(\mathbf{R_{xx}}) \tag{7.14}$$

Under assumptions (A1)-(A2), $[\mathbf{R_{xx}}]_{i,j} = \sigma_i^2\delta(i-j)$ where $\sigma_i^2 = \mathrm{E}\Big(|[\mathbf{x}]_i|^2\Big), i \in S_0$. Therefore, the non zero elements of $\mathrm{vec}(\mathbf{R_{xx}})$ are contained in the vector $\mathrm{diag}(\mathbf{R_{xx}}) \in \mathbb{R}^{N\times 1}$. It is readily seen that $\mathrm{Supp}\Big(\mathrm{diag}(\mathbf{R_{xx}})\Big) = S_0$. Using the definition of Khatri-Rao product, (7.14) thereby reduces to

(7.13) and this concludes the proof. □

The lemma suggests we can equivalently recover the sparse support if we can recover $\mathbf{p}$ satisfying (7.13). The problem formulation for recovery of $S_{\mathbf{0}}$ then becomes

$$\begin{aligned} &\min_{\mathbf{p}} ||\mathbf{p}||_0 \qquad\qquad (P0)_{Co-SMV} \\ &\text{subject to } \Big(\mathbf{A}^* \odot \mathbf{A}\Big)\mathbf{p} = \mathbf{z} \end{aligned} \tag{7.15}$$

Henceforth, we will denote the Khatri-Rao [100] product as

$$\mathbf{A}_{KR} = \mathbf{A}^* \odot \mathbf{A}$$

In order to understand what is gained by solving $(P0)_{Co-SMV}$ instead of $(P0)_{SMV}$, we define the following quantity called the "Degree of Krank Expansion" (denoted $\alpha_{\mathbf{A}}$) of $\mathbf{A}$:

Definition 7.2: The Degree of Krank Expansion of $\mathbf{A} \in \mathbb{C}^{M \times N}$ is defined as

$$\alpha_{\mathbf{A}} = \frac{\text{Krank}(\mathbf{A}_{KR})}{\text{Krank}(\mathbf{A})} \tag{7.16}$$

□

The quantity $\alpha_{\mathbf{A}}$ indicates if at all, using the "Co-SMV" framework, it is fundamentally possible to recover higher levels of sparsity than what is possible from the SMV model. This is formulated by the following lemma:

Lemma 7.5.2. *Let s_{SMV} and $s_{\text{Co-SMV}}$ denote the maximum size of S_0 that can be uniquely recovered by solving $(P0)_{SMV}$ and $(P0)_{Co-SMV}$ respectively. Then*

$$s_{\text{Co-SMV}} = \alpha_{\mathbf{A}} s_{\text{SMV}} \tag{7.17}$$

□

Proof. Applying Theorem 7.3.1 to $(P0)_{Co-SMV}$, we obtain that all sparse supports upto size

$$| S_0 | \leq \frac{\text{Krank}(\mathbf{A}_{KR})}{2}$$

can be uniquely recovered. The lemma then follows from definition of $\alpha_{\mathbf{A}}$. □

Hence, for $\alpha_{\mathbf{A}}$-krank expandable matrix $\mathbf{A}$, by solving $(P0)_{Co-SMV}$ we can recover sparsity levels which are $\alpha_{\mathbf{A}}$ times as large as the sparsity level that can be recovered by solving $(P0)_{SMV}$. The following theorem precisely characterizes how large this $\alpha_{\mathbf{A}}$ can be:

Theorem 7.5.1. *The Degree of Krank expansion of* $\mathbf{A}$ *satisfies*

$$1 < \alpha_{\mathbf{A}} \leq M \tag{7.18}$$

□

Proof. We use the following result from [164]

$$\text{Krank}(\mathbf{A} \odot \mathbf{B}) \geq \min(N, \text{Krank}(\mathbf{A}) + \text{Krank}(\mathbf{B}) - 1). \tag{7.19}$$

This implies $\text{Krank}(\mathbf{A}_{KR}) > \text{Krank}(\mathbf{A})$ and hence $\alpha_{\mathbf{A}} > 1$. Also let $\text{rank}(\mathbf{A}) = r$. From properties of Kronecker Products [91], $\text{rank}(\mathbf{A}^* \otimes \mathbf{A}) = r^2$. Now,

$$\text{Krank}(\mathbf{A}_{KR}) \leq \text{rank}(\mathbf{A}_{KR}) \leq \text{rank}(\mathbf{A}^* \otimes \mathbf{A}) = r^2$$

Hence, we obtain $\alpha_{\mathbf{A}} \leq r \leq M$. This concludes the proof. □

Theorem 7.5.1 implies the following:

1. Since, $\alpha_{\mathbf{A}} > 1$, we can always recover sparse support *of larger size* by using the "Co-SMV" framework.

2. Depending on $\mathbf{A}$, the maximum size of recoverable support can be M times larger than that obtained from solving $(P0)_{SMV}$, i.e., it can be $O(M^2)$.

This leads us to the following question:

Question: *"What are deterministic and random designs of* $\mathbf{A}$ *which can guarantee* $\text{KruskalRank}(\mathbf{A}_{KR}) = O(M^2)$*?"*

The following Theorem proves that random $\mathbf{A}$ with independently drawn entries has $\alpha_{\mathbf{A}} = O(M)$ almost surely:

Theorem 7.5.2. *Let* $\mathbf{A} \in \mathbb{R}^{M \times N}$ *be a random matrix whose elements are drawn* independently *from a continuous distribution over* $\mathbb{R}^{MN}$*. Assuming* $N > \frac{M^2+M}{2}$*, the following holds with probability* 1*:*

$$\mathrm{Krank}(\mathbf{A}_{\mathrm{KR}}) = \frac{M^2+M}{2} \tag{7.20}$$

□

Proof. We will show that any collection of $\frac{M^2+M}{2}$ columns in $\mathbf{A}_{KR}$ has rank $\frac{M^2+M}{2}$ with probability 1 over $\mathbb{R}^{\frac{M(M^2+M)}{2}}$. This will establish that $\mathbf{A}_{KR}$ has Kruskal Rank $\frac{M^2+M}{2}$ with probability 1 over $\mathbb{R}^{MN}$ by using the union bound over $\binom{N}{\frac{M^2+M}{2}}$ sets. Consider a subset $S \subseteq \{1, 2, \cdots, N\}$, where $\mid S \mid = \frac{M^2+M}{2}$ and let $[\mathbf{A}_{KR}]_S$ denote the submatrix consisting of columns indexed by the elements of S. Denote the elements of $\mathbf{A}$ as $a_{i,j}, \quad 1 \leq i \leq M, 1 \leq j \leq N$. The elements in the jth column of $\mathbf{A}_{KR}$ are given by $a_{i,j}a_{l,j}, \quad 1 \leq i, l \leq M$. Given integers $i_0, l_0 \in \{1, 2, \cdots, M\}, i_0 \neq l_0$,and $j \in S$ there are exactly two rows in $[\mathbf{A}_{KR}]_S$ that are identical and consist of the elements $a_{i_0,j}a_{l_0,j}, \quad 1 \leq j \leq N$. Overall, there are $\frac{M^2-M}{2}$ such identical rows, upon removing which we obtain an $\frac{M^2+M}{2} \times \frac{M^2+M}{2}$ submatrix of $[\mathbf{A}_{KR}]_S$ which we will denote as $[\mathbf{A}_{KR}]_S^{\mathsf{sub}}$. We will show that this matrix is full rank with probability 1. In order to establish this, denote $\mathbf{a}_S = \{a_{i,j}, 1 \leq i \leq M, j \in S\}$. Since the variables $a_{i,j}$ are drawn independently from a continuous distribution on $\mathbb{R}^{MN}$, the function $f(\mathbf{a}_S) \triangleq \det([\mathbf{A}_{KR}]_S^{\mathsf{sub}})$ is a multivariate polynomial in $M \mid S \mid$ variables given by the elements of $\mathbf{a}_S$ and hence it is analytic in $\mathbb{R}^{M|S|}$. By property of analytic functions, if $f(\mathbf{a}_S)$ is not the zero polynomial, its zero set will be of measure 0 in $\mathbb{R}^{M|S|}$. Given S, if we can prove that $f(\mathbf{a}_S)$ is not a trivial polynomial, then it will imply that the random matrix $[\mathbf{A}_{KR}]_S^{\mathsf{sub}}$ is full rank with probability 1 with respect to any continuous distribution over $\mathbb{R}^{M|S|}$ and it will conclude our proof. In order to prove $f(\mathbf{a}_S)$ is non trivial, it suffices to find a specific point $[\mathbf{a}_S]_0$ such that $f([\mathbf{a}_S]_0) \neq 0$. Such a $[\mathbf{a}_S]_0$ can be found by the following construction: Let $\{j_k, k = 1, 2, \cdots, \mid S \mid\}$ denote the indices included in S. Let us divide the $\frac{M^2+M}{2}$ columns indexed by S into M unequal groups $\{S_i\}_{i=1}^M$ with the $\mid S_m \mid = M - m + 1$. Also, relabel the columns in S_m as $\{j_k^m, 1 \leq k \leq M - m + 1\}$. The elements of $\mathbf{A}$ in columns indexed by $S_m, 1 \leq m \leq M$ are then given by:

$$a_{i,j_k^m} = \begin{cases} 1 & \text{if } i = m, k = 1 \\ 1 & \text{if } i = m, m+k-1, k > 1 \\ 0 & \text{otherwise} \end{cases} \tag{7.21}$$

The resulting matrix $[\mathbf{A}_S]^{\text{sub}}$ can be transformed, through row exchanges, to an upper triangular matrix, with each diagonal element being equal to 1. Hence it is full rank and this concludes the proof. □

The result shows that Khatri-Rao product of random matrices continue to exhibit very high Kruskal Rank. However, in many applications, such as imaging or source localization, the physics of the problem imposes structure on the measurement matrix $\mathbf{A}$. In many applications such as networks, the matrix $\mathbf{A}$ can represent the adjacency matrix of a network graph and can have further deterministic constraints. Usually in such situations, it is more difficult to come up with a deterministic design whose Khatri-Rao product continues to have $O(M^2)$ Kruskal Rank, and common sampling strategies often fail. In this context we would like to revisit the nested and coprime sampling strategies and show how these deterministic designs naturally possess $O(M^2)$ Kruskal Rank for their Khatri-Rao products.

7.5.1 Deterministic Designs: Revisiting Nested Arrays

Let us revisit the problem of direction finding in phased antenna arrays. Consider M antennas located at

$$z_i\lambda/2, \quad i = 1, 2, \cdots, M$$

where z_i are assumed integers. In DOA estimation problems, we assume D narrowband source signals impinging on this array of antennas from directions given by the set $\Theta = \{\theta_1, \cdots, \theta_D\}$. The lth time sample of the received signal vector (containing the signals at the M antennas) is given by

$$\mathbf{y}[l] = \mathbf{A}_{steer}(\Theta)\mathbf{s}[l], 1 \leq l \leq L \tag{7.22}$$

where $\mathbf{s}[k] \in \mathbb{C}^{D\times 1}$ denotes the source signal vector's lth time sample, $\mathbf{n}[k]$ denotes the vector of noise, assumed white and uncorrelated from signal. The source signals are assumed zero mean WSS random processes with power $\sigma_i^2, 1 \leq i \leq D$. The matrix $\mathbf{A}_{steer}(\Theta) \in \mathbb{C}^{M\times D}$ is the "Array Manifold Matrix" with elements

$$[\mathbf{A}(\Theta)]_{m,n} = e^{j\pi z_m \sin\theta_n} \tag{7.23}$$

To convert the problem to the linear sparse representation problem, following [121], we divide the range $[-\pi/2 \quad \pi/2)$ corresponding to all possible directions, into $N_\theta \gg N_{ca}$ grid points $\{\theta_1, \theta_2, \cdots, \theta_{N_\theta}\}$ and construct the (known) matrix $\mathbf{A}_{steer}^{grid} \in \mathbb{C}^{N_{ca} \times N_\theta}$ with columns corresponding to steering vectors for the candidate directions $\theta_i, 1 \leq i \leq N_\theta$. In particular, the elements of $\mathbf{A}_{sparse}^{grid}$ are given by

$$[\mathbf{A}_{sparse}^{grid}]_{m,n} = e^{j\pi z_m \sin\theta_n}, \quad m = 1, \cdots, M, \quad n = 1, \cdots, N_\theta \tag{7.24}$$

We can therefore approximate our model (7.22) as

$$\mathbf{y}[l] \approx \mathbf{A}_{steer}^{grid} \mathbf{s}_{sparse}[l] \tag{7.25}$$

The vector $\mathbf{s}_{sparse} \in \mathbb{C}^{N_\theta \times 1}$ is now a sparse vector with only D out of N_θ non zero (positive) values. Notice, by increasing N_θ, we can approximate (7.25) better. The unknown directions of arrival are now given by the support set of the vector $\mathbf{s}_{sparse}[l]$ and hence *direction finding becomes a sparse support recovery problem* with $\mathbf{A}_{steer}^{grid}$ acting as the *known* measurement matrix and the information about unknown directions contained entirely in the support of $\mathbf{s}_{sparse}[l]$.

It is to be noted that assumptions (A1) and (A2) naturally fit the context of direction finding because (A2) simply implies that the signals coming from different directions are statistically uncorrelated - an assumption that is commonly used. Denoting

$$\mathbf{A}^g = \mathbf{A}_{steer}^{grid}$$

we wish to understand how the structure of the array manifold can be exploited to ensure that the Kruskal Rank of $\mathbf{A}_{KR}^g$ is $O(M^2)$. We show that the conventional Uniform Linear Array (ULA) fails in this regard whereas coprime and nested arrays ensure the Kruskal Rank to be $O(M^2)$. To understand this, notice that the elements of $\mathbf{A}_{KR}^g$ are given by

$$e^{j\pi(z_i - z_j)\sin\theta_n}, \quad i, j \in \{1, 2, \cdots M\}, \quad n = 1, 2, \cdots N_\theta \tag{7.26}$$

The rows of $\mathbf{A}_{KR}^g$ are therefore characterized by the "difference set" as defined in Chapter 2:

$$S_{ca} = \{z_i - z_j, \quad 1 \leq i, j \leq M\} \tag{7.27}$$

The following lemma makes an explicit connection between the difference set and the Kruskal Rank of $\mathbf{A}_{KR}^{g}$:

Lemma 7.5.3. *The Kruskal Rank of* $\mathbf{A}_{KR}^{g}$ *satisfies*

$$\textit{Krank}(\mathbf{A}_{KR}^{g}) \leq \mid S_{ca} \mid \tag{7.28}$$

□

The number of distinct elements in S_{ca} therefore fundamentally limits the Krank of $\mathbf{A}_{KR}^{g}$. We now revisit the traditional Uniform Linear Array (ULA) and the nested array in the light of this fact. In each of the following cases, we assume that $N_g > M^2$:

1. **ULA**: For a uniform linear array,

   $$z_i = i, \quad 1 \leq i \leq M.$$

   It can be easily seen that in this case

   $$S_{ca} = \{m, -M+1 \leq m \leq M-1\}$$

   and hence $\mid S_{ca} \mid = 2M-1$. In fact, it can be verified that in this case Krank$(\mathbf{A}_{KR}^{g}) = 2M-1$. Since for a ULA, Krank$(\mathbf{A}_{steer}^{grid}) = M$, this shows from (7.19) that ULA actually achieves the minimum bound for the Kruskal Rank of its Khatri-Rao product.

2. **Nested Array**: For a nested array, the set of sensor locations is given by

   $$\{z_i\}_{i=1}^{M} = \{m, 1 \leq m \leq M/2\} \cup \{(M/2+1)n, 1 \leq n \leq M/2$$

   As shown in Chapter 2, the difference set is

   $$S_{ca} = \{m, -M^2/4 - M/2 + 1 \leq m \leq M^2/4 + M/2 - 1\}$$

and it can be verified that

$$\text{Krank}(\mathbf{A}_{KR}^{g}) = \frac{M^2}{2} + M - 1$$

Hence the nested array corresponds to a structured spatial sampling scheme that produces a deterministic measurement matrix with $\text{Krank}(\mathbf{A}_{KR}^{g}) = O(M^2)$.

It can be shown in a similar fashion that coprime arrays with M antennas can also give ensure that $\text{Krank}(\mathbf{A}_{KR}^{g}) = O(M^2)$.

7.6 Convex Relaxation for Support Recovery from Ideal Covariance Matrix

The preceding section established that it is possible to design both random and deterministic measurement matrices such that it becomes possible to identify supports of size $O(M^2)$ from the data covariance matrix $\mathbf{R_{yy}}$, say via exhaustive search. However, the practical approach to solve the support recovery problem is to solve a "relaxed" version:

$$\begin{aligned} &\min_{\mathbf{p}} ||\mathbf{p}||_1 \qquad\qquad (P1)_{Co-SMV} \\ \text{subject to} \quad &\Big(\mathbf{A}^* \odot \mathbf{A}\Big)\mathbf{p} = \mathbf{z} \end{aligned} \tag{7.29}$$

There are various ways to characterize when the solution to $(P0)_{Co-SMV}$ is identical to that of $(P1)_{Co_SMV}$. We will next describe a coherence based guarantee since the coherence for $\mathbf{A}$ and $\mathbf{A}_{KR}$ can be easily related.

7.6.1 New Coherence Based Guarantee

A sufficient condition that guarantees the equivalence of $(P0)_{Co-SMV}$ and $(P1)_{Co_SMV}$ problems is defined in terms of the coherence of the measurement matrix

Definition 7.3: The coherence of a matrix $\mathbf{A}$ is defined as

$$\mu_{\mathbf{A}} = \max_{i \neq j} \frac{| \mathbf{a}_i^H \mathbf{a}_j |}{\| \mathbf{a}_i \|_2 \| \mathbf{a}_j \|_2} \tag{7.30}$$

□

The following lemma relates the coherence of $\mathbf{A}$ and $\mathbf{A}^* \odot \mathbf{A}$:

Lemma 7.6.1. *The coherence of $\mathbf{A}_{KR}$ and $\mathbf{A}$ are related as:*

$$\mu_{\mathbf{A}_{KR}} = \mu_{\mathbf{A}}^2 \tag{7.31}$$

□

Proof. It can be verified from the definition of Kronecker products, that

$$(\mathbf{a}_i \otimes \mathbf{b}_i)^H (\mathbf{a}_j \otimes \mathbf{b}_j) = \Big(\mathbf{a}_i^H \mathbf{a}_j\Big)\Big(\mathbf{b}_i^H \mathbf{b}_j\Big)$$

Hence $\mid [\mathbf{a}_{KR}]_i^H [\mathbf{a}_{KR}]_j \mid = \mid (\mathbf{a}_i^* \otimes \mathbf{a}_i)^H (\mathbf{a}_j^* \otimes \mathbf{a}_j) \mid = (\mid \mathbf{a}_i^H \mathbf{a}_j \mid)^2$. This concludes the proof. □

The coherence based guarantee for finding the support S_0 by solving $(P1)_{Co-SMV}$ is given by

Theorem 7.6.1. *Consider the SMV model (7.1) where S_0 denotes the support of $\mathbf{x}$. Under assumptions (A1) and (A2), solving $(P1)_{Co-SMV}$ can uniquely recover supports of size*

$$\mid S_0 \mid < \frac{1}{2}\Big(1 + \frac{1}{\mu_{\mathbf{A}}^2}\Big) \tag{7.32}$$

□

Proof. It follows from the coherence based condition in [58] which states that if $\mathbf{y} = \mathbf{A}\mathbf{x}_0$ where the support of $\mathbf{x}_0$ is given by S_x, then solving $\min_{\mathbf{x}, \mathbf{A}\mathbf{x}=\mathbf{y}} \mid \mathbf{x} \mid_1$ yields $\mathbf{x}_0$ if $\mid S_0 \mid < \frac{1}{2}(1 + \frac{1}{\mu_A})$. □

Since, by definition $\mu_A < 1$, (7.31) implies that $\mu_{\mathbf{A}_{KR}} < \mu_{\mathbf{A}}$. Theorem 7.6.1 therefore shows that an argument based on coherence alone indicates that it is possible to recover larger supports by solving $(P1)_{Co-SMV}$. On one hand, coherence of a matrix is easy to compute which makes it a popular way to analyze recovery performance in many situations [23, 59, 179, 11]. On the other hand, it is only a sufficient condition for sparse recovery and suffers from the Welch bound [198], which indicates, for $\mathbf{A}_{KR}$,

$$\mu_{\mathbf{A}_{KR}} \geq \frac{1}{M}\sqrt{\frac{N - M^2}{N - 1}}. \tag{7.33}$$

For given N, this implies, in the Co-SMV forumalation, coherence based conditions on $\mathbf{A}_{KR}$ can guarantee recovery of supports of size only as large as $O(M)$. Hence coherence fails to indicate if it is at all possible to recover S_0 with $\mid S_0 \mid = O(M^2)$ by solving $(P1)_{Co-SMV}$. In what follows next, we will exploit additional structures in the Co-SMV model to provide much sharper guarantees.

7.7 The case of Positive Solution

In the previous sections, we ignored the fact that $\mathbf{p}$ contains the diagonal entries of $\mathrm{E}(\mathbf{x}\mathbf{x}^H)$ in (7.13) and is actually a non negative vector. In this section, we seek to recover the support S_0 by using this additional constraint on the unknown vector $\mathbf{p}$ to be non negative by solving

$$\begin{aligned} & \min_{\mathbf{p}} \|\mathbf{p}\|_1 && (P1+)_{Co-SMV} \\ \text{subject to} \quad & \mathbf{A}_{KR}\mathbf{p} = \mathbf{z}, \quad \mathbf{p} \succeq \mathbf{0} \end{aligned} \tag{7.34}$$

It is to be noted that even when the autocorrelation matrix is non ideal (i.e., estimated), the solution we are seeking is still non negative since it contains estimated power of the sources.

7.7.1 Conditions for Perfect recovery of non negative solution

The problem (7.34) has been of considerable interest [57, 61, 62] in compressive sensing. The necessary and sufficient conditions for perfect recovery of s-sparse non negative vectors relate to the theory of neighborly polytopes [62]. We shall first state a result (proved in [62]) when the measurement matrix is a Vandermonde matrix (with elements on the unit circle).

Theorem 7.7.1. *Let* $\mathbf{y} = \mathbf{A}\mathbf{x}_0$ *where the elements of* $\mathbf{A}$ *are given by* $[\mathbf{A}]_{m,n} = e^{jmt_n}, 1 \leq m \leq M, 0 < t_1 < \cdots < t_N < 2\pi$ *and* $\mathbf{x}_0 \succeq \mathbf{0}$*. If* $D \triangleq ||\mathbf{x}_0||_0 \leq M$ *and* $M < 2N$*, then* $\mathbf{x}_0$ *can be uniquely recovered by solving* $(P1+)_{Co-SMV}$*.* □

It is to be noted that the necessary and sufficient condition for $(P1+)_{Co-SMV}$ to recover the sparse positive vector *does not* involve a coherence or RIP condition on the measurement matrix $\mathbf{A}$. Rather, it requires the measurement matrix to be associated with a so-called outwardly $\mid S_0 \mid$-neighborly polytope to ensure perfect recovery. This particular structure of the measurement matrix is especially relevant in DOA estimation and makes it a perfect candidate for such applications as we

elaborate below.

7.7.2 Application to DOA estimation

Consider the direction finding problem revisited in Sec. 7.5.1 with the sparse representation given by (7.25). Traditionally, most DOA estimation algorithms exploit the correlation of the received data to estimate Θ. Assuming the source signals are uncorrelated, i.e., $E([\mathbf{s}]_i(t)[\mathbf{s}]_j^*(t)) = 0, i \neq j$, we compute $\mathbf{R}_{yy} \triangleq E(\mathbf{y}(l)\mathbf{y}(l)^H)$ and vectorize it to get

$$\mathbf{z} = \Big(\mathbf{A}_{steer}^{grid\ *} \odot \mathbf{A}_{steer}^{grid}\Big)\mathbf{p} + \sigma_n^2 \text{vec}\Big(\mathbf{I}\Big)$$

The vector $\mathbf{p}$ is a sparse vector with only D out of N_θ non zero (positive) values denoting the power of the D sources.

Denote N_{ca} as the number of distinct elements in the difference co-array S_{diff}. Extracting the elements in $\mathbf{z}$ corresponding to those distinct rows, we get the reduced sized vector

$$\mathbf{z}_1 = \mathbf{A}_{ca}\mathbf{p} + \sigma_n^2\mathbf{e}.$$

Since the difference co-array is always symmetric about 0, the vector $\mathbf{z}_1$ has an odd number of elements, the upper half being complex conjugates of the lower half. Also, $\mathbf{e}$ is a vector of all zeros except for a 1 in the middle. In order to remove the effect of noise, we only consider the upper $(N_{ca}-1)/2$ element subvector of $\mathbf{z}_1$ and get the model

$$\mathbf{z}_{1,u} = \mathbf{A}_{ca}^{+}\mathbf{p} \tag{7.35}$$

where rows of $\mathbf{A}_{ca}^{+} \in \mathbb{C}^{(N_{ca}-1)/2 \times N_\theta}$ correspond to elements of the difference co-array located at $\{z_i - z_j, 1 \leq i, j \leq M, z_i - z_j > 0\}$. Theorem 7.7.1 can be invoked to claim:

Theorem 7.7.2. *If $D \leq (N_{ca}-1)/2$, $N_{theta} \geq N_{ca}$, then the support of the positive sparse vector $\mathbf{p}$ in (7.35) can be recovered by solving (7.34) (using $\mathbf{z}_{1,u}$ in place of $\mathbf{z}$ and $\mathbf{A}_{ca}^{+}$ in place of $\mathbf{A}_{KR}$) provided the physical sensor locations satisfy*

$$S_{ca} = \{z_i - z_j, 1 \leq i, j \leq M\} = \pm\{0, 1, 2, \cdots, (N_{ca}-1)/2\} \tag{7.36}$$

□

In other words, the "difference set" of the physical array should contain a Uniform Linear Array (ULA) with N_{ca} elements. This brings us to the problem of designing physical arrays to achieve two goals:

1. (C1) The set S_{ca} should contain all possible integers from $(N_{ca}-1)/2$ to $(N_{ca}-1)/2$.
2. (C2) The integer N_{ca} must be as large as possible.

Evidently

$$N_{ca} \leq M(M-1)+1.$$

However, maximizing N_{ca} under the constraint (C1) is an open problem, with no analytical solution available for a given M. Following our discussion in Sec. 7.5.1, we can see that for the widely used ULA, $N_{ca} = 2M-1$ and that the nested array gives rise to $N_{ca} = O(M^2)$, while satisfying (C1). We therefore get the following Corollary to Theorem 7.7.2:

Corollary 7.7.1. *Consider the sparse signal model for direction finding (7.25) where* $\mathbf{A}_{steer}^{grid}$ *corresponds to the array manifold of a nested array. Then, by solving (7.34), it is possible to uniquely identify upto* $\frac{M^2}{4}+\frac{M}{2}-1$ *directions-of-arrival unambiguously from the support of the optimal solution* $\mathbf{p}^*$. □

The important points worth noticing about Corollary 7.7.1 are:

1. For nested arrays, the maximum recoverable sparsity by solving the $(P1+)_{Co-SMV}$ is *identical* to what could be obtained by solving $(P0)_{Co-SMV}$. Hence there is *no gap* in the recoverable sparsity limit in going from the non convex problem to the convex relaxation.
2. Conditions based on Coherence or the Restricted Isometry Property (RIP) lead to guarantees on the maximum size of recoverable $\mid S_0 \mid$ which depends on the number of columns N of $\mathbf{A}$ as $\mid S_0 \mid > \frac{M}{\log(N)}$

7.7.3 Simulations

Consider (i) a nested array, and (ii) a uniform linear array, both with $M = 6$ sensors. We perform DOA estimation by solving (7.34). Here $N_\theta = 800$, $SNR = 0$ dB. The nested array can identify upto 11 sources while the ULA can identify upto 5 sources. In contrast with the ideal autocorrelation

model discussed so far, we now *estimate* the autocorrelation matrix by averaging over T snapshots, i.e., $\hat{\mathbf{R}}_{yy} = \sum_{t=1}^{T} \mathbf{y}(t)\mathbf{y}(t)^H$.

Example 1: Assume $D = 5$ DOAs given by $-60°, -45°, -30°, -15°, 0°$. Here $D < M$, so both ULA and nested array can identify them using the correlation based model. We plot the recovered vector in Fig. 7.1(a)-(c) for both the arrays, for $T = 200$, by solving $(P1+)_{Co-SMV}$. It can be seen that the nested array can identify the 5 DOAs but the ULA fails at this value of snapshots. However, for $T = 800$ snapshots, it is able to identify the 5 DOAs. In order to demonstrate the superior identifiability of the nested array, we also plot (Fig. (7.1 (d)) the recovered vector for $D = 8(> M)$ sources impinging from $\{-60°, -45°, -30°, 0°, 30°, 36°, 45°, 60°\}$. Here $T = 400$. As can be seen, all 8 sources can be identified. We can also notice some smaller spurious peaks which indicates that the solution is only approximately sparse. This is due to the mismatch between the assumed ideal autocorrelation and its estimated value with finite snapshots.

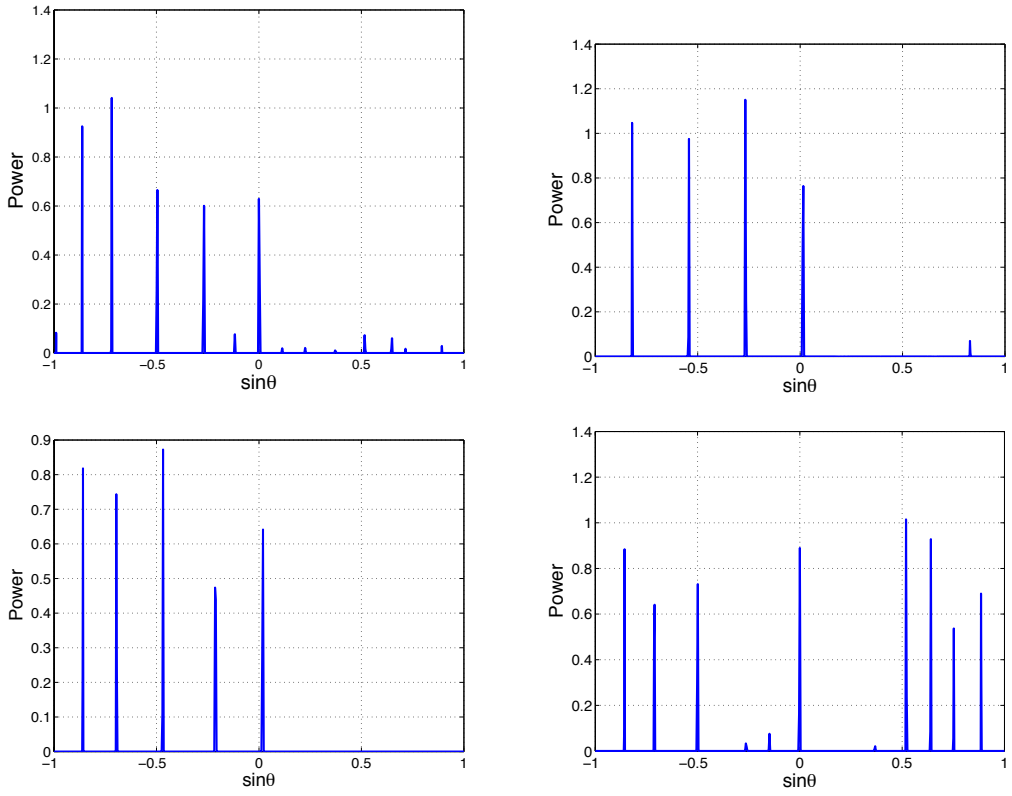


Figure 7.1: Comparison of recovered DOAs by nested array and ULA for different values of T (top left) Reconstructed DOAs using nested array with $D = 5$, $T = 200$. (top right) Reconstructed DOAs using ULA with $D = 5$, $T = 200$. (bottom left) Reconstructed DOAs using ULA with $D = 5$, $T = 800$, and (bottom right) Reconstructed DOAs using nested array with $D = 8$, $T = 400$.

Example 2: We now compare how well the nested array with 6 sensors can resolve two closely spaced sources compared to an ULA. We consider two closely spaced DOAs at 30° and 32°. The signals recovered by solving $(P1+)_{Co-SMV}$ are plotted for both the arrays in Fig. 7.2 for $T = 800$. The ULA is unable to resolve the two sources whereas the nested array can resolve them. This is because fundamentally the nested array, in the correlation based model, creates a much longer virtual array with "$O(M^2)$" new measurements and a larger aperture compared to the ULA with same number of elements.

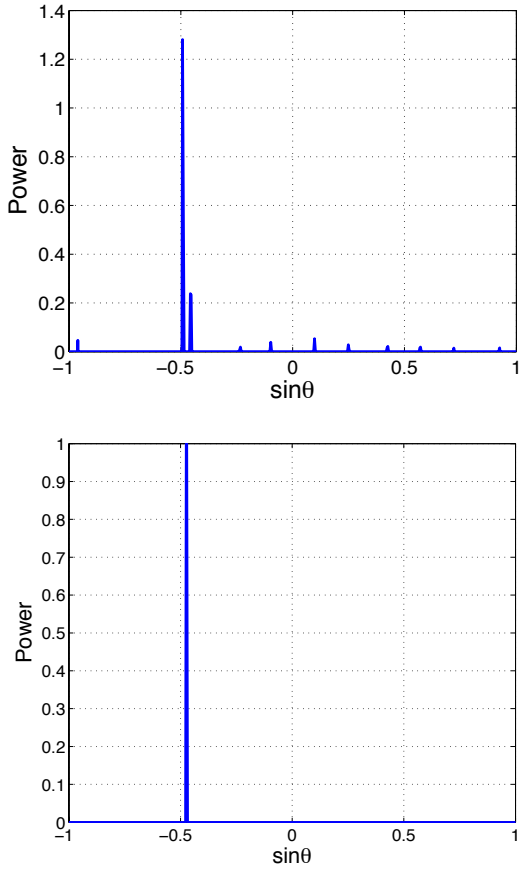


Figure 7.2: Comparison of resolution performance between nested array and ULA (top) Reconstructed DOAs using nested array for $D = 2$, $T = 200$, and (bottom) Reconstructed DOAs using ULA for $D = 2$, $T = 200$.

7.8 Orthogonal Matching Pursuit on the Covariance Matrix

Greedy algorithms provide an alternate route to the convex relaxation approaches for solving $(P0)$. The most popular among them is the Orthogonal Matching Pursuit (OMP) [52, 147, 177, 180]. In this section, we will provide new guarantees for OMP for our proposed Co-SMV model (7.13). In particular, we will show that it is possible to have significantly better recovery guarantees for OMP when applied on the Co-SMV model.

7.8.1 The Exact Reconstruction Condition for OMP

The condition which is necessary and sufficient for OMP to recover the true support S_0 is known as the Exact Reconstruction Condition (ERC) [177] and it is given by

Theorem 7.8.1. *[177] Given a sparse representation $\mathbf{y} = \mathbf{A}\mathbf{x}$ where x is s-sparse with S_0 denoting the* $\mathrm{Supp}(\mathbf{x})$*, the OMP will recover the columns of A indexed by S_0 in the first s steps if the following condition (also known as ERC) holds*

$$\max_{j \notin S_0} \| \left({\mathbf{A}_{S_0}}^H \mathbf{A}_{S_0} \right)^{-1} {\mathbf{A}_{S_0}}^H \mathbf{a}_j \|_1 < 1 \qquad (ERC) \tag{7.37}$$

□

The ERC condition can be very difficult to check. Two parameters of $\mathbf{A}$ that help to verify ERC are the coherence $\mu_{\mathbf{A}}$ and the cumulative coherence function $\mu_1^{\mathbf{A}}(s)$ as defined below:

Definition 7.4: [177] Given an integer s, the cumulative coherence function of a matrix $\mathbf{A}$ is defined as

$$\mu_1(s) = \max_{S:|S|=s} \max_{j \notin S} \| \mathbf{A}_S^H \mathbf{a}_j \|_1 \tag{7.38}$$

□

The relation of $\mu_{\mathbf{A}}$ and $\mu_1(s)$ to the ERC condition is given by the following theorem [177]

Theorem 7.8.2. *The condition ERC holds for every $S_0 \subset \{1, 2, \cdots, N\}$ with $|S_0| = s$ if*

$$s < \frac{1}{2}(1 + \frac{1}{\mu}) \tag{7.39}$$

or more generally

$$\mu_1^{\mathbf{A}}(s) + \mu_1^{\mathbf{A}}(s-1) < 1 \tag{7.40}$$

□

7.8.2 Guarantees for the Co-SMV Model

Recall that in the Co-SMV model, we have the matrix $\mathbf{A}_{KR}$ instead of $\mathbf{A}$. We already showed how $\mu_{\mathbf{A}}$ and $\mu_{\mathbf{A}_{KR}}$ are related. The following lemma relates $\mu_1^{\mathbf{A}_{KR}}(s)$ and $\mu_1^{\mathbf{A}}(s)$

Lemma 7.8.1. *Given an integer s, The cumulative coherence function of $\mathbf{A}$ and $\mathbf{A}_{KR}$ are related as*

$$\mu_1^{\mathbf{A}_{KR}}(s) < \Big(\mu_1^{\mathbf{A}}(s)\Big)^2 \tag{7.41}$$

□

Proof. Let $\mathbf{B}$ denote $\mathbf{A}_{KR}$ and $\mathbf{b}_i$ denote its ith column for ease of reference. Given $S \subseteq [1, 2, \cdots, N]$ with $\mid S \mid = s$,and integer $j \notin S$,

$$\|\mathbf{B}_S^H \mathbf{b}_j\|_1 = \sum_{i \in S} \mid \mathbf{b}_i^H \mathbf{b_j} \mid = \sum_{i \in S} \Big(\mid \mathbf{a}_i^H \mathbf{a}_j \mid \Big)^2 < \Big(\sum_{i \in S} \mid \mathbf{a}_i^H \mathbf{a}_j \mid \Big)^2 \leq \Big(\mu_1^{\mathbf{A}}(s)\Big)^2$$

Hence

$$\mu_1^{\mathbf{A}_{KR}}(s) = \max_{S:|S|=s} \max_{j \notin S} \|\mathbf{B}_S^H \mathbf{b}_j\|_1 < \Big(\mu_1^{\mathbf{A}}(s)\Big)^2$$

□

This leads to the following recovery condition for the Co-SMV framework:

Theorem 7.8.3. *The OMP algorithm applied on the Co-SMV model recovers the true support S_0 if*

$$s < \frac{1}{2}(1 + \frac{1}{\mu_{\mathbf{A}}^2}) \tag{7.42}$$

or more generally

$$\big(\mu_1^{\mathbf{A}}(s)\big)^2 + \big(\mu_1^{\mathbf{A}}(s-1)\big)^2 < 1 \tag{7.43}$$

□

An immediate corollary to Theorem 7.8.3 is given by

Corollary 7.8.1. *If*

$$\mu_1^{\mathbf{A}}(s) < \frac{1}{\sqrt{2}}, \tag{7.44}$$

then OMP on Co-SMV model can identify all supports of size s. □

Proof. It follows from Lemma 7.8.1. □

Notice that Corollary 7.8.1 strictly improves upon the condition in Corollary 3.6 in [177], according to which

$$\mu_1^{\mathbf{A}} < \frac{1}{2} \tag{7.45}$$

is a sufficient condition for OMP to recovery the true support. However, (7.45) implies (7.44) but not the converse. Denote s_{co-SMV} as the maximum integer s such that (7.44) holds and similarly let s_{SMV} be the maximum integer s such that (7.45) holds. Since $\mu_1^{\mathbf{A}}(s)$ is a non decreasing function of s [177], we can easily conclude that

$$s_{Co-SMV} \geq s_{SMV}$$

Hence the sufficient condition for success of OMP on the Co-SMV framework can guarantee recovery of sparsity levels which can be greater than that on the SMV model. Depending on the dictionary $\mathbf{A}$ itself, the difference between s_{Co-SMV} and s_{SMV} can be substantially large.

7.9 Conclusion

In this chapter, we proposed a correlation aware framework for sparse support recovery. Using priors on the unknown sparse signal which assume that the non zero entries are statistically uncorrelated, we showed that it is possible to have unique representation for the support even when the sparsity level exceeds the dimension of the measured vector(s). We developed new conditions based on the Khatri-Rao product of the measurement matrix to allow unique representation.

We also developed coherence based guarantees for a modified l_1 problem for support recovery from our proposed framework. The coherence based guarantees in this case turn out to be much stronger than existing counterparts in literature. We further exploited the non negativity of the solution and provided necessary and sufficient conditions for support recovery form the covariance matrix. These conditions were shown to naturally justify the choice of sparse sampling schemes such as nested and coprime arrays. Finally, we also considered greedy algorithms applied to the correlation aware framework and demonstrated that even for greedy approaches such as OMP, it is possible to have much stronger guarantees for support recovery by using an argument based on the cumulative coherence of the measurement matrix. The results developed in this chapter point to the effectiveness of our proposed formulation for recovering sparse supports of much larger size than what is considered in existing literature.

Chapter 8

Estimated Covariance matrix and the Multiple Measurement Vector Model

8.1 Introduction

The last chapter established new conditions on sparse recovery from the covariance matrix of the measurements. However, in practice, we only have estimate of the covariance matrix, computed from a finite amount of data. In this chapter, we focus on analyzing the performance of the covariance based recovery techniques when an estimate of the covariance matrix is used. In doing so, we will use the Multiple Measurement Vector (MMV) model (7.5) (which is a generalization of the SMV model) to obtain our estimate.

Jointly sparse support recovery from MMV is an important problem that seeks to recover the common support shared by the set of L sparse signal vectors $\mathbf{x}_l$ in (7.5). It arises in a number of applications such as MEG/EEG imaging [200, 117, 207, 208], wideband spectrum sampling [125], spectrum sensing for cognitive radio [153, 174, 203], direction finding in array processing [206, 172, 107, 121], radar[87], and multivariate regression [130, 129]. The MMV framework has been shown to provide fundamentally better guarantees [36, 49] than the SMV model in terms of uniqueness of sparse representation. Subsequently, a number of algorithms have been developed which solve the MMV problem for sparse recovery. These include greedy algorithms [181, 128, 48], convex relaxations via l_1/l_2 norm minimization and/or LASSO [178, 121, 36, 71, 130], hybrid of subspace based methods and convex relaxation [107], MMV extension of possibly non convex algorithms such as M-FOCUSS [49, 48] and Bayesian algorithms extended to the MMV model [200, 207]. Information theoretic limits for sparse recovery have also been studied [189, 172].

In this chapter, instead of assuming that we know the data covariance matrix, we estimate it from the MMV model using L measurements. As will be shown, this changes the formulation derived in Chapter 7 based on exact covariance matrix. In particular, as a result of estimation using finite number of samples, there is an additive noise like term which becomes arbitrarily small as $L \to \infty$ under our assumptions (A1)-(A2) on the correlation priors. We will devote this chapter to analyze the effect of such noise to the uniqueness condition of our model. We will also analyze the performance of LASSO [175] applied for support recovery from such a model.

8.1.1 Related Work and Our Contributions

Many of the algorithms for joint sparse recovery in MMV can be viewed as extensions of their SMV counterparts [181, 128, 48, 121, 178] and as such do not exploit any special structure or statistical prior of the measurements. However, there exists a family of algorithms which tries to exploit structure in the unknown sparse signals $\{\mathbf{x}_l\}_{l=1}^{L}$, such as rank [53, 107], correlation [207], non uniform sparse model with different probabilities on non zero entries [99] etc. Our work is more closely related to this second class of algorithms. However, the specific prior on correlation that we consider is very different from what has been considered in the literature so far. To the best of our knowledge, the only other (and possibly the first) body of work so far that considers correlation in the MMV model is [207, 208]. While the work in [207] considers the case where the L unknown vectors have temporal correlation and thereby proposes algorithms to learn the correlation by Sparse Bayesian Learning techniques, we actually consider the case when the non zero entries of the source vectors $\mathbf{x}[l]$ have *no correlation* among themselves. We have show that this assumption has significant implications in terms of design of suitable measurement matrices that can exploit such priors. The work in [207], however, does not specifically exploit the lack of correlation among the non zero elements, although it implicitly assumes a block diagonal covariance matrix for the matrix $\mathbf{X}_L$. Also, while the approach in [207] is based on SBL, we apply a LASSO on the covariance matrix of the data for recovering the sparse support. In future, we are interested in studying the effect of correlation priors in the SBL framework which has not been investigated so far.

The performance of LASSO for sparse models with additive noise has been analyzed [79, 190, 26, 59, 179, 123, 205]. In this chapter, we analyze LASSO for its capability to recover the underlying sparse support, as opposed to developing stability of the recovered vector in terms of l_2 norm

of the error. In this regard, our approach to analysis is closely related to that in [190, 79]. However, while [79] considers a bounded (non random) noise, we have a statistical model for our noise term, which is derived from the finite sample estimated covariance matrix. The analysis in [190] considers a random noise model and derives bounds on the probability of support recovery, both for deterministic and random ensemble of measurement matrices. They however consider a non random signal model for $\mathbf{X}_L$. In contrast, our formulation makes use of the second order moments of the distribution of a random signal model for $\mathbf{X}_L$. In our analysis, the measurement matrix is deterministic and the probability is measured over the distribution of the unknown signals. Consequently, our recovery guarantees are substantially different from those developed in [189] and we concentrate on how the probability of recovery behaves as L increases.

Portions from this chapter have been presented in [144] and [145].

8.2 Outline of Chapter

In Sec. 8.3, we formulate the problem of support recovery from an estimated covariance matrix and show the effect of finite measurements on such a model as an additive noise-like term. In Sec. 8.4, we establish uniqueness results for sparse representation using the proposed model. These guarantees are independent of any algorithm used. In Sec. 8.5, we propose the use of LASSO on the model derived from estimated covariance matrix and analyze its performance for any distribution of $\mathbf{X}_L$ which satisfy (A1) and (A2). Finally in Sec. 8.6, we compute bounds on the probability of successful support recovery when $\mathbf{X}_L$ follows multivariate Guassian distribution and show that it is possible to recover the support in this case with overwhelming probability as $L \to \infty$.

8.3 MMV Model and the Estimated Correlation Correlation

We revisit the MMV problem. Here $\mathbf{y}[l] \in \mathcal{R}^{M\times 1}, 1 \leq l \leq L$ denotes a set of L multiple measurement vectors, $\mathbf{A} \in \mathcal{R}^{M\times N} = [\mathbf{a}_1 \quad \mathbf{a}_2 \cdots \mathbf{a}_N]$ is the measurement matrix, and $\mathbf{x}_s[l] \in \mathcal{R}^{N\times 1}, 1 \leq l \leq L$ are L unknown vectors with the same sparsity $||\mathbf{x}_s[l]||_0 = D$ and common sparsity support denoted by the set $S_0 = \{i_0, i_1, \cdots, i_{D-1}\}$. We have

$$\mathbf{y}[l] = \mathbf{A}\mathbf{x}_s[l], \quad 1 \leq l \leq L. \tag{8.1}$$

We are interested in recovering the sparsity support S_0 using these ML measurements and determining what is the maximum D upto which all sparsity supports with cardinality D can be recovered. Without considering any priors on $\mathbf{X}_L$, one could only recover sparsity supports upto [36]

$$D < \frac{\text{Rank}(\mathbf{X}) + \text{Krank}(\mathbf{A}) - 1}{2}$$

which gives a maximum of $D < M$. In Chapter 7, it was demonstrated that with the perfect knowledge of the autocorrelation matrix of the vectors $\mathbf{y}[l]$, one can recover sparsity supports of much higher cardinality by applying Basis Pursuit (BP) on an appropriate model. The recoverable sparsity can be as high as $O(M^2)$ for suitably designed physical measurement matrices, which give rise to Vandermonde matrices in an appropriate model derived from the autocorrelation matrix. However, the formulation in Chapter 7 was developed for the ideal knowledge of the autocorrelation matrix of the measurement vectors $\mathbf{y}[l]$ which is true only in the limit at $L \to \infty$. In this paper, we consider the non ideal case, where we only have an estimate of the autocorrelation matrix from the L measurement vectors. We demonstrate that the non ideal knowledge of the covariance matrix manifests itself in the form of a random (possibly unbounded) additive noise term and hence the support recovery can be characterized only probabilistically.

The estimated sample correlation matrix is given by

$$\hat{\mathbf{R}}_{\mathbf{yy}}^{[L]} \triangleq \frac{1}{L}\sum_{l=1}^{L}\mathbf{y}[l]\mathbf{y}[l]^T = \mathbf{A}\hat{\mathbf{R}}_{xx}^{[L]}\mathbf{A}^T$$

where $\hat{\mathbf{R}}_{xx}^{[L]} = \frac{1}{L}\sum_{l=1}^{L}\mathbf{x}_s[l]\mathbf{x}_s[l]^T$ On vectorization, we can write

$$\mathbf{z}^{[L]} \triangleq \text{vec}(\mathbf{R}_{yy}^{[L]}) = \underbrace{(\mathbf{A}^* \odot \mathbf{A})}_{\mathbf{A}_{KR}}\mathbf{r}_{xx}^{[L]} + \mathbf{e}^{[L]} \tag{8.2}$$

where $\mathbf{r}_{xx}^{[L]} \in \mathcal{R}^{N\times 1}$ is a D sparse vector *with the same support* S_0 and non zero elements given by

$$\sigma_i^{2[L]} \triangleq \frac{1}{L}\sum_{l=1}^{L}[\mathbf{x}_s[l]]_i^2, i \in S_0 \tag{8.3}$$

Hence $\mathbf{r}_{xx}^{[L]}$ is a non negative vector for all values of L. The elements of the vector $\mathbf{e}^{[L]} \in \mathcal{R}^{M^2\times 1}$ are

given by

$$\frac{1}{L}\sum_{i\neq j}\sum_{l=1}^{L}\mathbf{A}_{m,i}\mathbf{A}_{n,j}[\mathbf{x}_s[l]]_i[\mathbf{x}_s[l]]_j, 1 \leq m,n \leq M. \tag{8.4}$$

They denote the estimates of cross correlation between the elements of $\mathbf{y}[l]$. The fact that the sample cross correlations are (unfortunately) nonzero for finite L is represented by these terms. We can think of $\mathbf{z}^{[L]}$ as our effective Single Measurement Vector (derived from the MMV), $\mathbf{A} \odot \mathbf{A}$ as the effective measurement matrix and $\mathbf{e}^{[L]}$ as the effective additive noise term. We aim to recover the support S_0 of the unknown vector $\mathbf{r}_{xx}^{[L]}$ from this model. Hence the imperfect knowledge of the ideal covariance matrix leads to an additive noise-like term $\mathbf{e}[L]$.

8.4 Fundamental Conditions for Sparse Support Recovery Under Bounded Noise

With only an estimate of the covariance matrix available (computed from finite number of samples), the problem of support recovery becomes identical to having an additive noise term to the Co-SMV model (7.13). Hence it is important to understand what conditions guarantee recovery of the true support under noisy setting. In other words consider a noisy observation

$$\mathbf{y} = \mathbf{A}\mathbf{x}_0 + \mathbf{n} \tag{8.5}$$

where $\mathrm{Supp}(\mathbf{x}_0) = S_0$ and $\| \mathbf{n} \|_2 < \epsilon$. We would be interested in knowing the condition under which the following (non convex) support recovery problem will yield the true support:

$$\min_{\mathbf{x}} \| \mathbf{x} \|_0 \qquad (P0)^{\epsilon} \tag{8.6}$$

$$\text{subject to} \| \mathbf{y} - \mathbf{A}\mathbf{x} \|_2 < \epsilon \tag{8.7}$$

Since ϵ gives a measure of the noise power, we define the Signal-to-Noise-Ratio (SNR) as

$$\text{SNR} \triangleq \frac{|\mathbf{x}_{0min}|}{\epsilon}. \tag{8.8}$$

Here $\mathbf{x}_{0_{min}}$ denotes the minimum non zero element of $\mathbf{x}_0$. We also define a parameter $\sigma_s(\mathbf{A})$ of $\mathbf{A}$ which will be useful to provide guarantees for recovery of S_0. This quantity is closely related to the "left RIP constant" as proposed by Candes [25].

Definition 8.1: Given a matrix $\mathbf{A}$ and integer s, the parameter $\sigma_s(\mathbf{A})$ is defined as

$$\sigma_s(\mathbf{A}) = \min_{S \subset [N]:|S|=s} \sqrt{\lambda_{\min}(\mathbf{A}_S^H \mathbf{A}_S)} \tag{8.9}$$

where for a matrix $\mathbf{B}$, $\lambda_{\min}(\mathbf{B})$ denotes the minimum eigenvalue. □

The following theorem states the sufficient condition under which $(P0)^\epsilon$ yield S_0.

Theorem 8.4.1. *Consider the model given by (8.5). The solution* $\mathbf{x}^*$ *to* $(P0)^\epsilon$ *satisfies* $\mathrm{Supp}(\mathbf{x}^*) = S_0$ *if*

$$\mathrm{Krank}(\mathbf{A}) > 2 \mid S_0 \mid \tag{8.10}$$

and

$$\mid \mathbf{x}_{0_{min}} \mid > \frac{2\epsilon}{\sigma_{2|S_0|}(\mathbf{A})}. \tag{8.11}$$

□

Proof. The solution $\mathbf{x}^*$ to $(P0)^\epsilon$ will recover the true support S_0 if and only if there is only one vector $\mathbf{x}_0$ with support S_0 that satisfies $\| \mathbf{y} - \mathbf{A}\mathbf{x}_0 \|_2 < \epsilon$. We will show that under the conditions $\mathrm{Krank}(\mathbf{A}) > 2 \mid S_0 \mid$ and $\mid \mathbf{x}_{0_{min}} \mid > \frac{2\epsilon}{\sigma_{2|S_0|}(\mathbf{A})}$, this is indeed true. We will prove this by contradiction. Let us assume that the solution $\mathbf{x}^*$ to $(P0)^\epsilon$ does not have the support S_0. Let $S_x = \mathrm{Supp}(\mathbf{x}^*)$. Since S_x is the minimizing support, $\mid S_x \mid \leq \mid S_0 \mid$. Since both $\mathbf{x}_0$ and $\mathbf{x}^*$ are feasible, we can write

$$\|\mathbf{A}_{S_x}[\mathbf{x}^*]_{S_x} - \mathbf{A}_{S_0}[\mathbf{x_0}]_{S_0}\|_2 < 2\epsilon \tag{8.12}$$

$$\Rightarrow \|\mathbf{A}_{S_0 \cup S_x}\mathbf{x}_T\|_2 < 2\epsilon \tag{8.13}$$

where $\mathbf{x}_T = \begin{bmatrix}[\mathbf{x}^*]_{S_x \setminus S_0} & [\mathbf{x}^*]_{S_x \cap S_0} - [\mathbf{x}_0]_{S_x \cap S_0} & -[\mathbf{x}_0]_{S_0 \setminus S_x}\end{bmatrix}^T$. Since $\mid S_x \mid < \mid S_0 \mid$, the number of columns in $\mathbf{A}_{S_0 \cup S_x}$ is at most $2 \mid S_0 \mid$ and we can say (8.13) implies

$$\sigma_{2|S_0|}(\mathbf{A}) \| \mathbf{x}_T \|_2 < 2\epsilon \tag{8.14}$$

Since Krank$(\mathbf{A}) > 2 \mid S_0 \mid$, $\sigma_{2|S_0|}(\mathbf{A}) > 0$ and we can say (8.14) implies

$$\| \mathbf{x}_T \|_2 < \frac{2\epsilon}{\sigma_{2|S_0|}(\mathbf{A})} \tag{8.15}$$

Since $S_0 \setminus S_x \neq \emptyset$, it follows that $\| \mathbf{x}_T \|_2 > \| [\mathbf{x}_0]_{S_0 \setminus S_x} \|_2 > \mid \mathbf{x}_{0min}$. | Therefore, (8.15) implies

$$\mid \mathbf{x}_{0min} \mid < \frac{2\epsilon}{\sigma_{2|S_0|}(\mathbf{A})}$$

which is a contradiction to the given condition, and this concludes our proof. □

Theorem 8.4.1 states that even in the noisy scenario, the following are true:

1. The solution of $(P0)^\epsilon$ can yield the true support S_0 of size upto $|S_0| < \frac{\text{Krank}(\mathbf{A})}{2}$. This number coincides with the maximum sized support obtainable by solving $(P0)$.
2. However, the SNR should exceed the threshold $t = \frac{2}{\sigma_{2|S_0|}(\mathbf{A})}$.

In the model (8.2) derived from the estimated correlation matrix, the matrix of interest is $\mathbf{A}_{KR}$. The following lemma establishes the relation between the singular values of $\mathbf{A}$ and $\mathbf{A}_{KR}$

Lemma 8.4.1. *Given a subset $S \subset \{1, 2, \cdots, N\}$, the minimum singular values of $\mathbf{A}_S$ and $\mathbf{A}_{KRS}$ satisfy*

$$\sigma_{min}(\mathbf{A}_{KRS}) \geq \sigma^2_{min}(\mathbf{A}_S) \tag{8.16}$$

□

Proof. From the definition of column-wise Kronecker products, it can be verified that

$$\begin{aligned} \mathbf{A}_{KRS}^H \mathbf{A}_{KRS} &= \left(\mathbf{A}_S^* \odot \mathbf{A}_S\right)^H \left(\mathbf{A}_S^* \odot \mathbf{A}_S\right) \\ &= \left(\mathbf{A}_S^T \mathbf{A}_S^*\right) \circ \left(\mathbf{A}_S^H \mathbf{A}_S\right) \end{aligned}$$

From properties of Hadamard Product, for two positive semidefinite matrices $\mathbf{M}, \mathbf{N}$, the minimum eigenvalues λ_{min} satisfy [92, p. 312]

$$\lambda_{min}(\mathbf{M} \circ \mathbf{N}) \geq \lambda_{min}(\mathbf{M}) \lambda_{min}(\mathbf{N}) \tag{8.17}$$

from which the desired condition follows. □

The problem of recovering the sparsest $\mathbf{r}_{xx}^{[L]}$ satisfying (8.2) can be cast as

$$\min_{\mathbf{r}} \| \mathbf{r} \|_0 \qquad (P0)_{Co-SMV}^{epsilon} \tag{8.18}$$

$$\text{subject} \quad \text{to} \quad \| \mathbf{z}^{[L]} - \mathbf{A}_{KR}\mathbf{r} \|_2 \leq \epsilon \tag{8.19}$$

where ϵ is an upper bound on the error of estimation $\|\mathbf{e}^{[L]}\|_2$.

Applying Theorem (8.4.1), we get the sufficient condition for the solution of $(P0)_{Co-SMV}^{\epsilon}$ to uniquely recover the support S_0:

Theorem 8.4.2. *Consider the model (8.2) derived from the MMV model with ML measurements. The solution $\mathbf{r}^*$ to $(P0)_{Co-SMV}^{\epsilon}$ satisfies* $\text{Supp}(\mathbf{r}^*) = S_0$ *if*

$$\text{Krank}(\mathbf{A}_{KR}) > 2 \mid S_0 \mid \tag{8.20}$$

and

$$\mid \mathbf{r}_{xx\,min}^{[L]} \mid > \frac{2\epsilon}{\sigma_{2|S_0|}(\mathbf{A}_{KR})} \tag{8.21}$$

□

The theorem in particular implies

1. It is fundamentally possible to uniquely identify sparse support of size as large as Krank$(\mathbf{A}_{KR})$. Since Krank$(\mathbf{A}_{KR}) \geq 2$Krank$(\mathbf{A})$, it is fundamentally possible to recover larger sparsity levels if the SNR is high enough.

2. The above claim is true if the SNR condition $\mid \mathbf{r}_{xx\,min}^{[L]} \mid > \frac{2\epsilon}{\sigma_{2|S_0|}(\mathbf{A}_{KR})}$ holds. For Support S_0 with $\mid S_0 \mid < \frac{\text{Krank}(\mathbf{A})}{2}$, which can be recovered by solving both $(P0)^{\epsilon}$ and $(P0)_{Co-SMV}^{\epsilon}$, the SNR threshold required for $(P0)_{Co-SMV}^{\epsilon}$ can be higher than that for $(P0)^{\epsilon}$ if $\sigma_{2|S_0|}(\mathbf{A}) < 1$. This is evident from Lemma 8.4.1.

Hence by solving the non convex problem $(P0)_{Co-SMV}^{\epsilon}$ derived from the proposed correlation aware framework, it is possible to recover larger sparse support provided the SNR is large enough. However, for smaller sparsity levels, the SNR requirement can be significantly larger than what is

needed for solving $(P0)^{\epsilon}$. Since $(P0)^{\epsilon}_{Co-SMV}$ is NP-hard, we next analyze the performance of the convex relaxation of $(P0)^{\epsilon}_{Co-SMV}$.

8.5 Analysis of LASSO: Independent of Distribution

Since the imperfect knowledge of the ideal covariance matrix leads to an additive noise-like term $\mathbf{e}[L]$, this provides the ideal framework for applying LASSO (l_1 norm regularized quadratic programming) [175] for sparse support recovery. Since the signal vector is always positive, in our case, we have an added constraint on the unknown vector to be positive. So the problem of recovering the sparse support becomes finding the support of a vector $\hat{\mathbf{r}} \in \mathcal{R}^{N\times 1}$ which is the solution to the following constrained LASSO:

$$\begin{aligned} &\min_{\mathbf{r}} \Big(\frac{1}{2}||(\mathbf{A}\odot\mathbf{A})\mathbf{r} - \mathbf{z}^{[L]}||_2^2 + h||\mathbf{r}||_1\Big) \qquad (LASSO) \\ &\text{subject to } \mathbf{r} \succeq \mathbf{0} \end{aligned} \tag{8.22}$$

.

In [79, 190], the performance of LASSO for support recovery has been analyzed for both bounded and sub Gaussian noise (for deterministic as well as stochastic measurement matrix $\mathbf{A}$). However, unlike [79, 190], we do not assume the effective additive noise term $\mathbf{e}^{[L]}$ to be bounded or sub Gaussian. We only have assumptions on the second order statistics of the noise term (which results from our original assumption about uncorrelatedness of the vectors) and not on their specific distributions. With this lone assumption, we compute a lower bound on the probability of successful recovery of the support. As a first step, the following theorem provides a set of sufficient conditions for the solution of the LASSO (8.22) to yield the true support of the sparse vectors:

Theorem 8.5.1. *Consider the model (8.2) where S_0 denotes the sparse support of $\mathbf{r}_{xx}^{[L]}$. If $\mid S_0 \mid < \frac{1}{2}(1+\frac{1}{\mu_A^2})$, and the following two conditions hold:*

$$\|\mathbf{e}^{[L]}\|_2 < h\frac{1+\mu_A^2-2\mu_A^2 \mid S_0 \mid}{1+\mu_A^2-\mu_A^2 \mid S_0 \mid}, \qquad (E1) \tag{8.23}$$

$$\sigma_i^{2^{[L]}} > \frac{\|\mathbf{e}^{[L]}\|_2 + h}{1 + \mu_A^2 - \mu_A^2 \mid S_0 \mid}, \quad \forall i \in S_0 \qquad (E2) \tag{8.24}$$

then the optimal solution $\mathbf{r}^*$ *to the LASSO satisfies* $\mathrm{Supp}(\mathbf{r}^*) = S_0$. □

Proof. See Appendix 8.9.1. □

Notice that the above set of sufficient conditions involves the random variables $\sigma_i^{2^{[L]}}$ and $\mathbf{e}^{[L]}$. Since E_1 and E_2 are sufficient conditions for support recovery, if we can develop a lower bound on the probability of joint occurrence of the events E_1 and E_2, then the probability of successful support recovery by solving LASSO can also be lower bounded by the same quantity. Denote

$$\mu_i^{(4)} \triangleq E([\mathbf{x}_s[l]]_i^4) \quad \text{and} \quad \sigma_{min}^2 = \min\{\sigma_i^2, i \in S_0\}$$

Lemma 8.5.1. *Consider the model (8.2) and let (A1) and (A2) hold. If* $0 < h < \sigma_{min}^2 \frac{(1+\mu_A^2-\mu_A^2|S_0|)^2}{2(1+\mu_A^2)-3\mu_A^2|S_0|}$*, the probability of the event* $E_1 \bigcap E_2$ *is bounded below by*

$$\mathcal{P}(E_1 \bigcap E_2) > 1 - \frac{C(h)}{L} + O(\frac{1}{L^2})$$

where the constant $C(h)$ *is independent of* L *and depends on* h *as well as on coherence* μ_A*, sparsity* D *and the matrix* $\mathbf{A}$ *as*

$$C(h) = \sum_{i=1}^{D} \frac{\mu_i^{(4)} - \sigma_i^4}{(\sigma_i^2 - h\frac{2(1+\mu_A^2)-3\mu_A^2 D}{1+\mu_A^2-\mu_A^2 D})^2} + \frac{M^2(1+\mu_A^2-\mu_A^2 D)^2}{h^2(1+\mu_A^2-2\mu_A^2 D)^2} \sum_{i\neq j}^{D} \sigma_i^2\sigma_j^2 \Big(\|\mathbf{a}_i\|_2^2 \|\mathbf{a}_j\|_2^2 + (\mathbf{1}_{M\times 1}^T [\mathbf{a}_i \circ \mathbf{a}_j])^2 \Big)$$

□

Proof. See Appendix 8.9.4. □

Combining Theorem 8.5.1 and Lemma 8.5.1, we obtain the following theorem which sums up the desired result on the probability of successful recovery:

Theorem 8.5.2. *Consider the MMV model (8.1) with the assumptions given by (A1)-(A2). If*

$$\mid S_0 \mid < \frac{1}{2}(1 + \frac{1}{\mu_A^2}), \quad \textit{and} \quad 0 < h < \sigma_{min}^2 \frac{(1 + \mu_A^2 - \mu_A^2 \mid S_0 \mid)^2}{2(1+\mu_A^2) - 3\mu_A^2 \mid S_0 \mid},$$

then the common support S_0 can be recovered by solving LASSO given by (8.22), with probability greater than $1 - \frac{1}{L}C(h) + O(\frac{1}{L^2})$ □

Following points are worth noticing:

1. When $L \to \infty$, one can guarantee the recovery of any sparsity level upto $\frac{1}{2}(1 + \frac{1}{\mu_A^2})$ with probability 1. Since we used our assumption only on the second order statistics of the random vectors (and not their distribution), the above probability of successful recovery was bounded using Chebyshev Inequality. Consequently, the upper bound on probability of failure decays as $\frac{1}{L}$ which can be rather loose for smaller values of L. However, with more assumptions on the distribution of nuisance terms (such as sub Gaussian tail etc), the bound can be tightened by using Chernoff-like bounds (as done in [190]). It is also to be noted that unlike most analysis (including that in [190]) on sparse recovery which considers probability of successful recovery over an ensemble of random measurement matrices, the current analysis has been performed for a fixed measurement matrix $\mathbf{A}$ and the probability is considered with respect to the distribution of the random source vectors $\mathbf{x}_s[l]$. Also, our goal is to characterize how it varies with L.

2. *Role of cross correlation*: It is evident from above analysis that the cross correlation between $[\mathbf{x}_l]_i$ and $[\mathbf{x}_l]_j, \quad i \neq j$ has an additive noise like effect on the ideal model considered in Chapter 7. These non zero cross correlation terms arise from the finite number of snapshots and their norm decreases with high probability as L increases. Hence, we can bound their norm at any given value of L with a probability that increases with L, and this in turn dictates the probability of successful recovery using LASSO. A similar analysis may also be extended to the case where the entries in $\mathbf{x}_l$ are actually correlated and one can derive interesting relations between the SNR and the allowable magnitudes of these cross correlation terms for the success of LASSO.

8.6 Analysis of LASSO For Gaussian Model

In the previous section, we performed analysis of the LASSO (8.22) by computing the probability of successful support recovery without assuming any specific probability distribution for the source vectors. Accordingly, the lower bound on the probability of successful recovery (denoted $\mathcal{P}_s$) was found to be rather loose, and it increases with L as $\mathcal{P}_s \geq 1 - \frac{C}{L}$. As evident, the bound increases

to unity rather slowly with increasing L. In this section, we demonstrate that assuming a specific distribution on the unknown source vectors, namely, multivariate Gaussian, it is possible to guarantee support recovery with overwhelming probability with increasing L. In particular, we will show that the probability of support recovery increases as $\mathcal{P}_s \geq 1 - C\beta^{-L}, \quad C > 0, \beta > 1$.

8.6.1 Concentration Inequalities

We first state a concentration inequality ([86], Lemma 6) that will be directly used in our derivation:

Lemma 8.6.1. *Let each of x_i and y_i, $i = 1, \cdots, k$ be real sequences of i.i.d zero mean Gaussian random variables with variance σ_x^2 and σ_y^2 respectively. Then*

$$\mathcal{P}\Big(|\sum_{i=1}^{k} x_i y_i| \geq t\Big) \leq 2\exp\Big(-\frac{t^2}{2\sigma_x\sigma_y(2\sigma_x\sigma_y k + t)}\Big)$$

□

Using above lemma, we can get the following result. The notation $\sigma_{max}^{(k)}$ denotes the kth largest element in the set of non negative numbers $\{\sigma_i\}_{i=1}^{|S_0|}$.

Lemma 8.6.2. *Consider the model (8.2). Under the assumptions (A1)-(A2), the following holds $\forall 1 \leq i \leq M^2$:*

$$\mathcal{P}\Big(|[\mathbf{e}^{[L]}]_i| \geq C\Big) \leq 2\exp\Big(-\frac{Lt^2}{2\sigma_{max}^{(1)}\sigma_{max}^{(2)}(2\sigma_{max}^{(1)}\sigma_{max}^{(2)} + t)}\Big),$$

where $t \triangleq \frac{C}{\|\mathbf{A}\|_{\infty,\infty}^2 |S_0|(|S_0|-1)}$ □

Proof. See Appendix 8.9.2 □

We now prove a second concentration inequality which directly provides a bound on the probability of occurrence of event E_2.

Lemma 8.6.3. *Let $x_i, i = 1, \cdots, L$ denote i.i.d zero mean Gaussian random variables with variance σ_x^2. Also, assume $0 < C < \sigma_x^2$. Then, there exists $\beta > 1$ such that*

$$\mathcal{P}\Big(\frac{1}{L}\sum_{i=1}^{L} x_i^2 > C\Big) \geq 1 - \beta^{-L} \tag{8.25}$$

□

Proof. See Appendix 8.9.3 □

8.6.2 Probability of Support recovery by Solving the LASSO

Armed with the inequalities provided by Lemmas 8.6.2 and 8.6.3, we now state our main result on the probability of support recovery by solving the LASSO (8.22), given by the following theorem:

Theorem 8.6.1. *Consider the MMV model given by (8.1) which satisfies the assumptions (A1)-(A2). If*

$$\mid S_0 \mid < \frac{1}{2}(1+\frac{1}{\mu_A^2}), \quad \textit{and} \quad 0 < h < \sigma_{min}^2 \frac{(1+\mu_A^2-\mu_A^2 \mid S_0 \mid)^2}{2(1+\mu_A^2)-3\mu_A^2 \mid S_0 \mid},$$

then the common support S_0 can be recovered by solving the proposed LASSO given by (8.22), with probability greater than $1-\alpha\gamma^{-L}$ for some $\gamma > 1$. □

Proof. Using Theorem 8.5.1, we can say that the probability of successful recovery of sparse support by solving the LASSO (8.22), denoted as P_s, satisfies:

$$P_s \geq \mathcal{P}(E_1 \bigcap E_2) \tag{8.26}$$

where $(E1)$ and $(E2)$ are defined in (8.23) and (8.24). Using a technique similar to the proof of Lemma 8.5.1 in Appendix 8.9.4, it can be shown that

$$\mathcal{P}(E_1 \bigcap E_2) \geq \prod_{i=1}^{D} \mathcal{P}(\sigma_i^{2[L]} > c_2) - \sum_{i=1}^{M^2} \mathcal{P}(\mid [\mathbf{e}^{[L]}]_i \mid \geq \frac{c_1}{M}) \tag{8.27}$$

Here c_1 and c_2 are as defined in Appendix 8.9.4 Now using the expression for $\sigma_i^{2[L]}$ from (8.3), we can say, from Lemma 8.6.3 that

$$\prod_{i=1}^{|S_0|} \mathcal{P}(\sigma_i^{2[L]} > c_2) \geq \prod_{i=1}^{|S_0|} (1-\beta_i^{-L}) \tag{8.28}$$

and using Lemma 8.6.2, we get

$$\sum_{i=1}^{M^2} \mathcal{P}\Big(\mid [\mathbf{e}^{[L]}]_i \mid \geq \frac{c_1}{M} \Big) \leq 2M^2 e^{-\delta L} \tag{8.29}$$

where $\delta \triangleq \frac{t^2}{2\sigma_{max}^{(1)}\sigma_{max}^{(2)}(2\sigma_{max}^{(1)}\sigma_{max}^{(2)}+t)}$ and $t \triangleq \frac{c_1}{M\|\mathbf{A}\|_{\infty,\infty}^2 |S_0|(|S_0|-1)}$. Using (8.28) and (8.29) in (8.64), we

obtain

$$\mathcal{P}_s \geq \prod_{i=1}^{|S_0|}(1-\beta_i^{-L}) - 2M^2 e^{-\delta L} \tag{8.30}$$

which proves the desired result since each $\beta_i > 1$. □

8.7 Numerical Results

Consider $\mathbf{A} \in \mathcal{R}^{M\times N}$ to be a Gaussian matrix with zero mean and unit variance. We first assume $M = 20, \quad N = 256$ and the sparsity level is $|\ S_0\ | = 34$. We consider $L = 200$ i.i.d Gaussian source vectors, and apply our proposed technique (correlation aware LASSO) as well as existing variants of MMV BP [36, 121] on the same data.

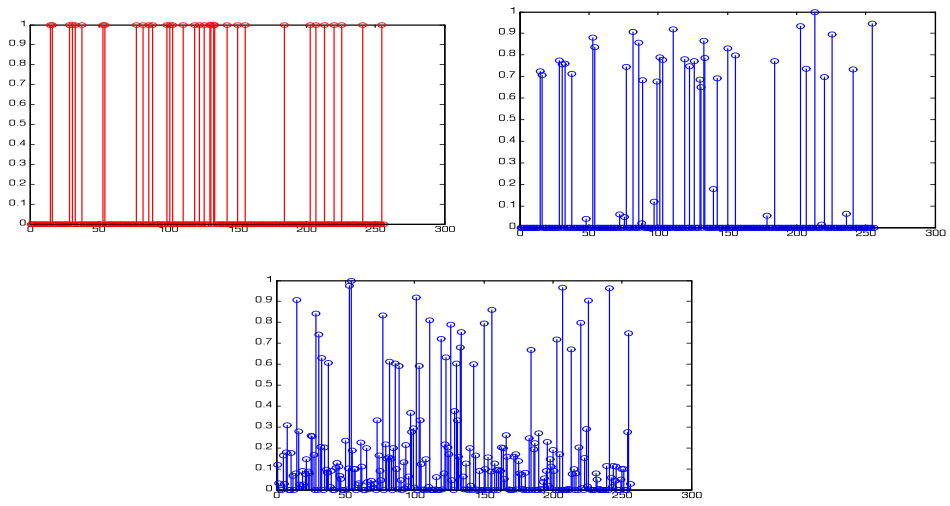


Figure 8.1: Comparison of recovered support when $|\ S_0\ | = 34 > M$. (top left): True Support. (Top right): Support recovered by LASSO on Co-SMV, (bottom) Support recovered by MMV-BP. Here $M = 20, N = 256$.

Fig. 8.1 compares the performance of these two methods for sparsity level $|\ S_0\ | = 34$ by plotting the recovered vector. It can be seen that the proposed method outperforms traditional MMV BP which fails to identify the correct support for sparsity levels larger than M.

Next, we demonstrate the probability of successful support recovery (by generating zero mean i.i.d. random $\mathbf{x}_l, \quad 1 \leq l \leq L$ but keeping $\mathbf{A}$ constant) averaged over 500 Monte Carlo runs. We consider $M = 20, N = 256$ and two different levels of sparsity levels.

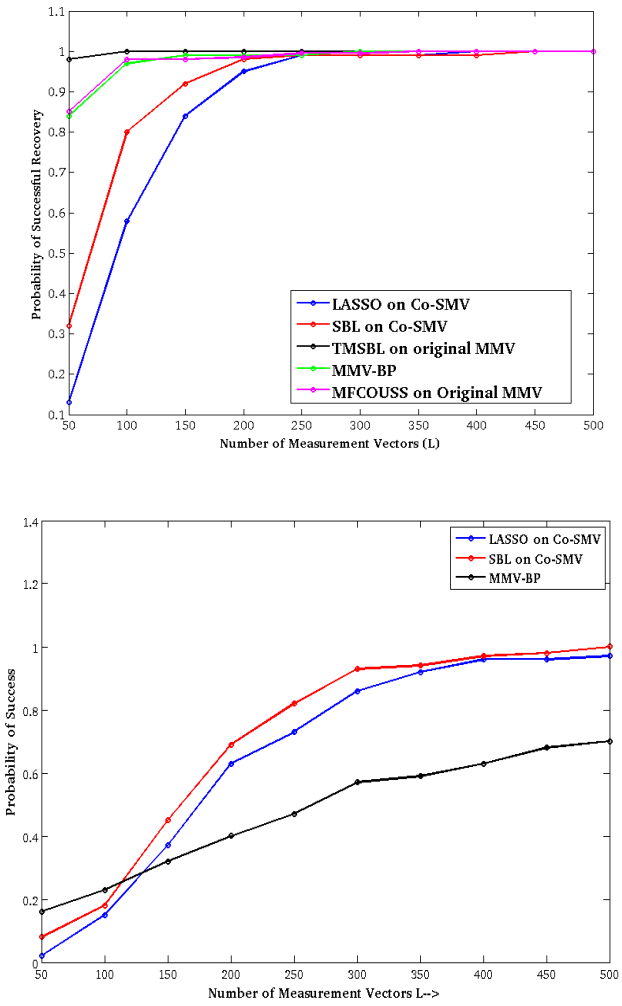


Figure 8.2: Probability of Support recovery in two different regimes: (top) $\mid S_0 \mid < M$ and (bottom) $\mid S_0 \mid > M$.

We first plot the probability of successful recovery for $\mid S_0 \mid = 16 < M$ in Figure 8.2 (top). In this scenario, many existing MMV sparse recovery algorithms are capable of recovering the sparse support and we provide a comparison with a number of algorithms. In particular, we compare with TMSBL [207], MFOCUSS [49, 48] and MMV-BP. It can be seen that for large L, all the algorithms can recover the true support with probability 1. However, for small L, the existing approaches outperform SBL and LASSO applied on the proposed framework. This is because for small L, the estimates of the correlation have the non zero cross correlation terms added and they create the effect of noise. The traditional methods do not consider estimates of correlation and hence do not suffer from this effect.

In the second case, we consider $\mid S_0 \mid = 24 > M$. We plot the probability of successful support recovery vs L for two algorithms, namely, SBL [176] and LASSO [175] applied to our proposed Co-SMV framework. We also compare the performance of MMV-BP in this case. As seen earlier, the solution of MMV-BP is not even sparse in this scenario. Hence to permit a comparison, we run MMV-BP under the assumption that the sparsity level $\mid S_0 \mid = 24$ is known and hence it picks the 24 largest peaks from it non sparse solution. However, the two algorithms applied on our Co-SMV framework do not have this side information. Figure 8.2 (bottom) shows the desired plot. It can be seen that the proposed framework outperforms MMV-BP by a significant margin. This proves the effectiveness of the proposed framework for dealing with sparsity levels larger than M.

8.8 Conclusion

In this chapter, we studied the performance of correlation aware sparse support recovery when we only have estimates of correlation obtained from a finite number (L) of measurements from the MMV model. It was shown that the effect of finite measurements has a noise-like effect on the model for support recovery and the use of LASSO was proposed. We analyzed the performance of LASSO for support recovery by probabilistically characterizing the effect of estimated correlation values. We first performed a general analysis without considering any specific distribution for $\mathbf{x}$ and then extended this analysis to the case where $\mathbf{x}$ belongs to a multivariate Gaussian distribution with diagonal covariance matrix. We showed that it is possible to recover the support with a probability which grows overwhelmingly fast with L.

8.9 Appendix

8.9.1 Proof of Theorem 8.5.1

We introduce Lagrange multipliers λ_i, $1 \leq i \leq N$ corresponding to the constraints $-\mathbf{r} \preceq \mathbf{0}$, with $\lambda_i \geq 0$ for each i. Using the KKT conditions [18], we obtain that ($\mathbf{r}^*$,$\boldsymbol{\lambda}^*$) constitute a primal-dual optimal solution pair if and only if

$$\Big(\mathbf{A}^{(KR)^T}\mathbf{A}^{(KR)}\Big)\mathbf{r}^* - \mathbf{A}^{(KR)^T}\mathbf{z}^{[L]} + h\mathbf{1}_{N\times 1} - \boldsymbol{\lambda}^* = 0 \tag{8.31}$$

and

$$[\boldsymbol{\lambda}^*]_i[\mathbf{r}^*]_i = 0, 1 \leq i \leq N \tag{8.32}$$

Here $\mathbf{A}^{(KR)} = \mathbf{A} \odot \mathbf{A}$. Recall that S_0 denotes the set containing the indices indicating the true support of the sparse vector $\mathbf{r}_{xx}^{[L]}$ that generates the observed $\mathbf{z}^{[L]}$. Given a vector $\mathbf{v}$, and a set S of integers, let us denote $\mathbf{v}_S$ as the $|S| \times 1$ subvector of $\mathbf{v}$ containing elements of $\mathbf{v}$ whose indices are given by the set S. Similary for a matrix $\mathbf{A}$, $\mathbf{A}_S$ is the submatrix composed of $|S|$ columns indexed by S. With this notation, it can be seen that $\hat{\mathbf{r}}$ and $\hat{\boldsymbol{\lambda}}$ as defined below (by (8.33)-(8.36)), are an optimal primal-dual pair since they can be verified to satisfy (8.31) and (8.32).

$$\hat{\mathbf{r}}_{S_0^c} \triangleq \mathbf{0}, \tag{8.33}$$

$$\hat{\boldsymbol{\lambda}}_{S_0} \triangleq \mathbf{0} \tag{8.34}$$

$$\hat{\mathbf{r}}_{S_0} \triangleq \mathbf{r}_{xx\,S_0}^{[L]} + \mathbf{A}_{S_0}^{(KR)^\dagger}\mathbf{e}^{[L]} - h\Big(\mathbf{A}_{S_0}^{(KR)^T}\mathbf{A}_{S_0}^{(KR)}\Big)^{-1}\mathbf{1}_{S_0} \tag{8.35}$$

$$\hat{\boldsymbol{\lambda}}_{S_0^c} \triangleq h\mathbf{1}_{S_0^c} - \mathbf{A}_{S_0^c}^{(KR)^T}\Big(\mathbf{e}^{[L]^\perp} + h\mathbf{A}_{S_0}^{(KR)^{\dagger T}}\mathbf{1}_{S_0}\Big) \tag{8.36}$$

Here $S_0^c = [N] \setminus S_0$ and $\mathbf{e}^{[L]^\perp} = \mathbf{e}^{[L]} - \mathbf{A}_{S_0}^{(KR)}\mathbf{A}_{S_0}^{(KR)^\dagger}\mathbf{e}^{[L]}$. Now, if h is such that:

$$h(1-\mathbf{a}_i^{(KR)^T}\mathbf{A}_{S_0}^{(KR)^{\dagger^T}}\mathbf{1}_{S_0}) > \mathbf{a}_i^{(KR)^T}\mathbf{e}^{[L]^\perp}, i \in S_0^c \tag{8.37}$$

then it implies from (8.36)

$$[\hat{\boldsymbol{\lambda}}]_i > 0, \forall i \in S_0^c \tag{8.38}$$

This condition has an important implication in recovery of true support. Any optimal $\mathbf{r}^*$ must now satisfy (8.32) with the particular optimal dual variable $\hat{\boldsymbol{\lambda}}$ satisfying (8.38) and this immediately implies: $\mathrm{Supp}(\mathbf{r}^*) \subseteq S_0$

We assume from now on that (8.37) holds true. Using this fact, for any primal optimal $\mathbf{r}^*$ and the particular dual optimum $\hat{\boldsymbol{\lambda}}$ given by (8.34) and (8.38), we obtain from (8.31) that

$$\mathbf{r}^*_{S_0} = \hat{\mathbf{r}}_{S_0} \tag{8.39}$$

where $\hat{\mathbf{r}}_{S_0}$ is as in (8.35).

Now if $\mathrm{Supp}(\mathbf{r}^*) \subset S_0$ (i.e., the true support is not exactly equal to the support of the optimal vector) then $\exists i$ such that

$$|[\mathbf{r}^*_{S_0} - \mathbf{r}^{[L]}_{xx}]_i| \geq {\sigma^2_{min}}^{[L]} \triangleq \min_{j \in S_0}\{\sigma_j^{2[L]}\}$$

Conversely, we can say from (8.35), if

$$\|\mathbf{A}_{S_0}^{(KR)^\dagger}\mathbf{e}^{[L]} - h\Big(\mathbf{A}_{S_0}^{(KR)^T}\mathbf{A}_{S_0}^{(KR)}\Big)^{-1}\mathbf{1}_{S_0}\|_\infty < {\sigma^2_{min}}^{[L]} \tag{8.40}$$

then it implies that

$$\mathrm{Supp}(\mathbf{r}^*) = S_0.$$

In conclusion, we can say that the support of any optimal solution of the LASSO (8.22) will be equal to the true support S_0 provided (8.37) and (8.40) hold.

Following an approach similar to [79], it can be shown that if $D < \frac{1}{2}(1 + \frac{1}{\mu_A^2})$, then (8.37) is satisfied provided

$$\|\mathbf{e}^{[L]}\|_2 < h\frac{1+\mu_A^2-2\mu_A^2 D}{1+\mu_A^2-\mu_A^2 D} \tag{8.41}$$

and (8.40) is satisfied if

$$\sigma_{\min}^{2\ [L]} > \frac{\|\mathbf{e}^{[L]}\|_2 + h}{1+\mu_A^2-\mu_A^2 D} \tag{8.42}$$

Summarizing, if (8.23) and (8.24) hold, then

$$\mathrm{Supp}(\mathbf{r}^*) = S_0$$

for any optimal $\mathbf{r}^*$.

8.9.2 Proof of Theorem 8.6.2

Notice from (8.4) that

$$|[\mathbf{e}^{[L]}]_i| \leq \frac{\| \mathbf{A} \|_{\infty,\infty}^2}{L} \sum_{k\neq j} \sum_{l=1}^{L} \left| [\mathbf{x}_s[l]]_k [\mathbf{x}_s[l]]_j \right| \tag{8.43}$$

Therefore

$$|[\mathbf{e}^{[L]}]_i| \geq C \Longrightarrow \sum_{k\neq j} \sum_{l=1}^{L} \left| [\mathbf{x}_s[l]]_k [\mathbf{x}_s[l]]_j \right| \geq \frac{CL}{\| \mathbf{A} \|_{\infty,\infty}^2} \tag{8.44}$$

Also, notice that

$$\sum_{k\neq j} \sum_{l=1}^{L} \left| [\mathbf{x}_s[l]]_k [\mathbf{x}_s[l]]_j \right| \geq \frac{CL}{\| \mathbf{A} \|_{\infty,\infty}^2} \Longrightarrow \tag{8.45}$$

$$\sum_{l=1}^{L} \left| [\mathbf{x}_s[l]]_{k_0} [\mathbf{x}_s[l]]_{j_0} \right| > \frac{CL}{\| \mathbf{A} \|_{\infty,\infty}^2 D(D-1)} \tag{8.46}$$

for some $k_0, j_0 \in \{1, \cdots, D\}, k_0 \neq j_0$. Using (8.44) and (8.46), we can say that

$$\mathcal{P}\Big(|[\mathbf{e}^{[L]}]_i| \geq C\Big) \tag{8.47}$$

$$\leq \mathcal{P}\Big(\sum_{l=1}^{L} \Big|[\mathbf{x}_s[l]]_{k_0}[\mathbf{x}_s[l]]_{j_0}\Big| > \frac{CL}{\|\mathbf{A}\|^2_{\infty,\infty} D(D-1)}\Big) \tag{8.48}$$

Under assumption (A1), $\{[\mathbf{x}_s[l]]_j\}_{l=1}^L$ and $\{[\mathbf{x}_s[l]]_k\}_{l=1}^L$ denote sequences of i.i.d. zero mean Gaussian random variables which are independent of each other. Hence, given $k \neq j$, we can use Lemma 8.6.1 to obtain

$$\mathcal{P}\Big(\sum_{l=1}^{L} \Big|[\mathbf{x}_s[l]]_{k_0}[\mathbf{x}_s[l]]_{j_0}\Big| > \frac{CL}{\| \mathbf{A} \|^2_{\infty,\infty} D(D-1)}\Big) \tag{8.49}$$

$$\leq 2\text{exp}\Big(-\frac{Lt^2}{2\sigma_{k_0}\sigma_{l_0}(2\sigma_{k_0}\sigma_{l_0}+t)}\Big) \tag{8.50}$$

$$\leq 2\text{exp}\Big(-\frac{Lt^2}{2\sigma^{(1)}_{max}\sigma^{(2)}_{max}(2\sigma^{(1)}_{max}\sigma^{(2)}_{max}+t)}\Big) \tag{8.51}$$

The desired result directly follows from above and (8.48).

8.9.3 Proof of Lemma 8.6.3

The proof is based upon Chernoff Bound. Denote $p_x^{[L]} \triangleq \frac{1}{L}\sum_{i=1}^{L} x_i^2$. Then

$$\mathcal{P}(p_x^{[L]} > C) = \mathcal{P}\Big(\sum_{i=1}^{L} x_i^2 > CL\Big) = \mathcal{P}\Big(\sum_{i=1}^{L} z_i^2 > \frac{CL}{\sigma_x^2}\Big) \tag{8.52}$$

where z_i denote i.i.d zero mean standard normal variables. Therefore $\sum_{i=1}^{L} z_i^2$ is a Chi-Squared random variable with L degrees of freedom. Observe $\text{P}(\text{p}_x^{[L]} > C) = 1 - \mathcal{P}\Big(\sum_{i=1}^{L} z_i^2 \leq \frac{CL}{\sigma_x^2}\Big)$ $\geq 1 - \exp(\frac{sCL}{\sigma_x^2})(1+2s)^{-L/2}$ for $s > 0$. Equation (8.9.3) follows from the Chernoff Bound and also from the fact that the Moment Generating function of a Chi-Squared random variable with L degrees of freedom is given by $(1-2s)^{-L/2}, s < 1/2$. Now define the function

$$\beta(s) = \exp(-\frac{2sC}{\sigma_x^2})(1+2s) \tag{8.53}$$

we can write from (8.9.3) that

$$\mathcal{P}(p_x^{[L]} > C) \geq 1 - \Big(\beta(s)\Big)^{-L/2}. \tag{8.54}$$

We want to ensure that $\exists s > 0$, such that $\beta(s) > 1$. Now $\beta(s) > 1 \Leftrightarrow C < \gamma(s)$ where $\gamma(s) \triangleq \frac{\sigma_x^2}{2s}\log(1+2s)$. It can be verified that $\gamma(s)$ is a decreasing function in s for $s > 0$ and $\gamma(0) = \sigma_x^2$. Since, it is given that $C < \sigma_x^2$, then indeed $\exists s_0 > 0$ such that $C < \gamma(s_0)$. This in turn implies, $\beta(s_0) > 1$. Hence, we conclude from (8.54) that $\mathcal{P}\Big(\frac{1}{L}\sum_{i=1}^{L} x_i^2 > C) \geq 1 - \beta^{-L}$ for $\beta = \sqrt{\beta(s_0)} > 1$.

8.9.4 Proof of Lemma 8.5.1

The probability of successful recovery of sparse support by solving the LASSO (8.22), denoted as P_s, satisfies:

$$P_s \geq \mathcal{P}(E_1 \bigcap E_2) \tag{8.55}$$

$$= \mathcal{P}\Big(\|\mathbf{e}^{[L]}\|_2 < h\frac{1+\mu_A^2-2\mu_A^2 \mid S_0 \mid}{1+\mu_A^2-\mu_A^2 \mid S_0 \mid}, {\sigma_{\min}^2}^{[L]} > \frac{\|\mathbf{e}^{[L]}\|_2 + h}{1+\mu_A^2-\mu_A^2 D}\Big) \tag{8.56}$$

If (E_1) holds, then ${\sigma_{\min}^2}^{[L]} > \frac{h\frac{1+\mu_A^2-2\mu_A^2 D}{1+\mu_A^2-\mu_A^2 D}+h}{1+\mu_A^2-\mu_A^2 D} \Rightarrow {\sigma_{\min}^2}^{[L]} > \frac{\|\mathbf{e}^{[L]}\|_2+h}{1+\mu_A^2-\mu_A^2 D}$. Using this, we get

$$P_s \geq \mathcal{P}\Big(\|\mathbf{e}^{[L]}\|_2 < c_1, {\sigma_{\min}^2}^{[L]} > c_2\Big) \tag{8.57}$$

where

$$c_1 \triangleq h\frac{1+\mu_A^2-2\mu_A^2 D}{1+\mu_A^2-\mu_A^2 D}, \quad c_2 \triangleq h\frac{2(1+\mu_A^2)-3\mu_A^2 D}{(1+\mu_A^2-\mu_A^2 D)^2} \tag{8.58}$$

Now

$$P_s \geq \mathcal{P}(\|\mathbf{e}^{[L]}\|_2 < c_1, {\sigma_{\min}^2}^{[L]} > c_2) \tag{8.59}$$

$$\geq \mathcal{P}\Big(\bigcap_{i=1}^{M^2}\Big\{ \mid [\mathbf{e}^{[L]}]_i \mid < \frac{c_1}{M}\Big\}, {\sigma_{\min}^2}^{[L]} > c_2\Big) \tag{8.60}$$

$$= \mathcal{P}({\sigma_{\min}^2}^{[L]} > c_2) - \mathcal{P}({\sigma_{\min}^2}^{[L]} > c_2, \bigcup_{i=1}^{M^2}\Big\{ \mid [\mathbf{e}^{[L]}]_i \mid \geq \frac{c_1}{M}\Big\}) \tag{8.61}$$

$$\geq \mathcal{P}({\sigma_{\min}^2}^{[L]} > c_2) - \mathcal{P}(\bigcup_{i=1}^{M^2}\Big\{ \mid [\mathbf{e}^{[L]}]_i \mid \geq \frac{c_1}{M}\Big\}) \tag{8.62}$$

$$\geq \mathcal{P}({\sigma_{\min}^2}^{[L]} > c_2) - \sum_{i=1}^{M^2}\mathcal{P}(\mid [\mathbf{e}^{[L]}]_i \mid \geq \frac{c_1}{M}) \tag{8.63}$$

$$= \prod_{i=1}^{D} \mathcal{P}(\sigma_i^{2^{[L]}} > c_2) - \sum_{i=1}^{M^2} \mathcal{P}(\mid [\mathbf{e}^{[L]}]_i \mid \geq \frac{c_1}{M}) \tag{8.64}$$

Now, recall from (8.4) that each element $\mathbf{e}_k^{[L]}$ of the vector $\mathbf{e}^{[L]}$ is of the form

$$\frac{1}{L} \sum_{l=1}^{L} \sum_{i \neq j} \mathbf{A}_{m,i} \mathbf{A}_{n,j} [\mathbf{x}_s[l]]_i [\mathbf{x}_s[l]]_j$$

for some $m, n \in \{1, 2, \cdots, M\}$. Therefore, $E(\mathbf{e}_k^{[L]}) = \frac{1}{L} \sum_{l=1}^{L} \sum_{i \neq j} \mathbf{A}_{m,i} \mathbf{A}_{n,j} E\Big([\mathbf{x}_s[l]]_i [\mathbf{x}_s[l]]_j\Big) = 0$ due to assumption (A1). Furthermore, using (A1) it can be shown that

$$E({\mathbf{e}_k^{[L]}}^2) = \frac{1}{L} \sum_{i \neq j} \Big(\mathbf{A}_{m,i}^2 \mathbf{A}_{n,j}^2 + \mathbf{A}_{m,i} \mathbf{A}_{m,j} \mathbf{A}_{n,i} \mathbf{A}_{n,j}\Big) \sigma_i^2 \sigma_j^2 \tag{8.65}$$

If $h > 0$ which implies $c_1 > 0$, applying Chebyshev's Inequality, we obtain

$$\mathcal{P}(\mid \mathbf{e}_k^{[L]} \mid \geq \frac{c_1}{M}) \leq \frac{E({\mathbf{e}_k^{[L]}}^2) M^2}{c_1^2} \tag{8.66}$$

Substituting (8.65) in (8.66), we obtain

$$\sum_{k=1}^{M^2} \mathcal{P}(\mid \mathbf{e}_k^{[L]} \mid \geq \frac{c_1}{M}) \leq \frac{M^2}{c_1^2} \sum_{k=1}^{M^2} E({\mathbf{e}_k^{[L]}}^2) \tag{8.67}$$

$$= \frac{M^2}{c_1^2 L} \sum_{i \neq j} \sigma_i^2 \sigma_j^2 \Big(\|\mathbf{a}_i\|_2^2 \|\mathbf{a}_j\|_2^2 + (\mathbf{1}^T [\mathbf{a}_i \circ \mathbf{a}_j])^2\Big) \tag{8.68}$$

On the other hand, from (8.3), we obtain

$$E(\sigma_i^{2^{[L]}}) = \sigma_i^2$$

Under the assumption of the theorem $h < \sigma_{min}^2 \frac{(1+\mu_A^2-\mu_A^2 D)^2}{2(1+\mu_A^2)-3\mu_A^2 D}$, we have

$$\sigma_i^2 > c_2 = h \frac{2(1+\mu_A^2) - 3\mu_A^2 D}{(1+\mu_A^2-\mu_A^2 D)^2}, \quad 1 \leq i \leq D$$

Hence we can say that

$$\mathcal{P}(\sigma_i^{[L]^2} > c_2) \geq \mathcal{P}(\mid \sigma_i^{[L]^2} - \sigma_i^2 \mid < \sigma_i^2 - c_2) \tag{8.69}$$

Applying Chebyshev Inequality, we get

$$\mathcal{P}(\mid \sigma_i^{[L]^2} - \sigma_i^2 \mid \geq \sigma_i^2 - c_2) \leq \frac{\mu_i^{(4)} - \sigma_i^4}{L(\sigma_i^2 - c_2)^2} \tag{8.70}$$

since $E\Big(\sigma_i^{2[L]} - E(\sigma_i^{2[L]})\Big)^2 = \frac{1}{L}(\mu_i^{(4)} - \sigma_i^4)$.

From (8.69) and (8.70), we get

$$\mathcal{P}(\sigma_i^{[L]^2} > c_2) \geq 1 - \frac{\mu_i^{(4)} - \sigma_i^4}{L(\sigma_i^2 - c_2)^2} \tag{8.71}$$

Using (8.68) and (8.71) in (8.64), we obtain the desired bound on the probability of success.

Chapter 9

Future Directions

In this thesis, we considered underdetermined estimation problems and established the role of priors in designing efficient sampling and reconstruction techniques, both for some typical problems in array processing and broadly to more general sparse observation models. The new perspectives offered in this thesis open up a number of future directions which are summarized below.

1. **Measurement matrix design in the new formulation** : Our proposed approach to sparse support recovery has opened up a host of new questions regarding measurement matrix design. In our formulation, the support recovery algorithm works not on the physical measurement matrix $\mathbf{A}$ but on the derived matrix $\mathbf{A}_{\mathrm{KR}}$. One question of interest is: *how to design the physical measurement matrix such that* $\mathbf{A}_{\mathrm{KR}}$ *will have desirable recovery properties*. For example, one can ask the question: how to design $\mathbf{A}$ such that $\mathbf{A}_{\mathrm{KR}}$ would satisfy well known sufficient conditions for sparse recovery such as Restricted Isometry property (RIP), Exact Reconstruction Condition (ERC) etc. The Null Space Condition which is *both necessary and sufficient* for sparse recovery, also needs to be revisited in the context of our new formulation.

2. **Sparse recovery and distribution of the data** : In our framework for sparse support recovery (inspired from co-array), we have mostly exploited the correlation (i.e. second order statistics) among the multiple measurement vectors. However, if we have more assumptions on the *distribution of the measurement vectors*, we can possibly utilize that knowledge to find better strategies for sparse support recovery which can guarantee the recovery of even higher levels of sparsity, as we had demonstrated in the context of DOA estimation using *higher order statistics*. These may include formulation of the sparse support recovery in a different domain (derived from higher order moments of the distribution), by computing appropriate statistics

from the data. Some specific questions that can be asked in this regard are:

- Characterize the probability of successful support recovery and investigate if one can recover $O(s^2)$ sparsity levels with overwhelming probability (i.e., of the form $1 - e^{-CL}$, L being the number of measurement vectors and C being a positive constant) using assumptions on the distribution of the measurement vectors.
- Investigate how the knowledge of higher order moments of the distribution can increase the sparsity levels.
- Establish information theoretic bounds and study the fundamental limits on sparse recovery from an observation model which is equipped with prior information.

This can lead to finding new structures for the measurement matrices given the knowledge of the distribution of the data.

3. **Learning of the dictionary/measurement matrix** : In many situations where we are unaware of the basis in which the data could be sparse, the approach is to learn the dictionary (and the associated sparse representation) from the data itself [170]. This is often formulated as an iterative optimization approach. We can ask a similar question in the setting of our problem based on the derived SMV from the autocorrelation of the data. However there is a difference due to the fact that our effective dictionary in this correlation domain is different from the physical measurement matrix. This gives rise to a very interesting dictionary learning problem where dictionary is now *constrained to have the specific structure* (related to the Khatri Rao product as discussed earlier) that arises in our proposed co-array based framework. This added constraint in the iterative optimization procedure can give rise to new approaches towards learning of such dictionaries. Various tools from machine learning (such as neural networks) can also be employed to find the underlying dictionary. Potential applications can be for representation of large data sets arising in computer vision, genomics, web search, geophysical data, remote sensing, social networks and so forth.

4. **Sparse Representation for Learning** : Signal representation theory has been one of the cornerstones of signal processing, and the recent advances in compressive sensing has provided new perspectives into it by exploiting sparsity in the representation. Such representation can be very useful for high dimensional signals such as image, video, seismic data, and more

generally data arising from a variety of sources such as web search, database queries, social network, stock prices, and so forth. The theory of learning tends to derive meaningful inference from these high dimensional data for a variety of tasks such as prediction, detection, recognition, classification, etc. The unsupervised learning approach often aims at finding a sparse or low dimensional representation of the data, which can also be termed as "feature extraction", where the features belong to a lower dimensional space. Recent advances in machine learning have been in the direction of investigating if such a sparse representation can be efficiently found from large training datasets [202]. One of the existing approaches is to use sparse autoencoders [88] which employs neural networks to learn the representation. By *blending tools from classical and modern signal processing with those in machine learning*, new insights can be gained regarding feasibility of such sparse representation and its role in learning. A plausible future direction would be to investigate into learning algorithms that exploit the statistical distribution of the underlying data while seeking a sparse representation. It will be interesting to see if the work presented in this thesis can be extended to such problems.

Bibliography

[1] Y.I. Abramovich, D. A. Gray, and A. Y. Gorokhov, N. K. Spencer, "Positive-definite Toeplitz completion in DOA estimation for nonuniform linear antenna arrays. I. Fully augmentable arrays," *IEEE Transactions on Signal Processing*, vol.46, no.9, pp. 2458-2471, Sep 1998.

[2] Y. I. Abramovich, N. K. Spencer, and A. Y. Gorokhov, "Positive-definite Toeplitz completion in DOA estimation for nonuniform linear antenna arrays. II. Partially augmentable arrays," *IEEE Transactions on Signal Processing*, vol. 47, no. 6, pp. 1502-1521, Jun 1999.

[3] Y. I. Abramovich, N. K. Spencer, A. Y. Gorokhov, "Detection-estimation of more uncorrelated Gaussian sources than sensors in nonuniform linear antenna arrays - part III: detection-estimation nonidentifiability", *IEEE Transactions on Signal Processing*, vol. 51, no. 10, pp. 2483- 2494, Oct. 2003.

[4] M. Akcakaya and V. Tarokh, "A frame construction and a universal distortion bound for sparse representations, *IEEE Transactions on Signal Processing* vol. 56, no. 6, pp. 2443-2450, Jun. 2008.

[5] M. Akcakaya and V. Tarokh, "Shannon theoretic limits on noisy compressive sampling", *IEEE Transactions on Information Theory*, vol. 56, no. 1, pp. 492-504, 2010.

[6] A. Amar, A.J. Weiss, "Fundamental Limitations on the Number of Resolvable Emitters Using a Geolocation System," *IEEE Transactions on Signal Processing*, vol. 55, no. 5, pp. 2193-2202, May 2007.

[7] W. U. Bajwa, J. Haupt, A. M. Sayeed, and R. Nowak, "Compressed channel sensing: A new approach to estimating sparse multipath channels", *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1058-1076, 2010.

[8] R. G. Baraniuk, "More is less: Signal processing and the data deluge", *Science*, vol. 331, no. 6018, pp. 717-719, 2011.

[9] R. G. Baraniuk, V. Cevher, M. F. Duarte, and C. Hegde, "Model-based compressive sensing" *IEEE Transactions on Information Theory*, vol. 56, no. 4, pp. 1982-2001, April 2010.

[10] R. G. Baraniuk, M. Davenport, R. A. DeVore, and M. Wakin. "A simple proof of the restricted isometry property for random matrices" *Constructive Approximation*, vol. 28, pp. 253-263, 2008.

[11] Z. Ben-Haim, Y. C. Eldar, and M. Elad, "Coherence-based performance guarantees for estimating a sparse vector under random noise", *IEEE Transactions on Signal Processing* vol. 58, no. 10, pp. 5030-5043, Oct. 2010.

[12] E. van den Berg and M. Friedlander, "SPGL1: A solver for large-scale sparse reconstruction" 2007. [Online]. Available: http://www.cs.ubc.ca/labs/scl/spgl1.

[13] E. Van Den Berg and M. Friedlander, "Theoretical and empirical results for recovery from multiple measurements, *IEEE Transactions on Information Theory*, vol. 56, no. 5, pp. 2516-2527, May 2010.

[14] G. Birot, L. Albera, P. Chevalier, "Sequential High-Resolution Direction Finding From Higher Order Statistics," *IEEE Transactions on Signal Processing*, vol. 58, no. 8, pp. 4144-4155, Aug. 2010.

[15] J. Blanchard, C. Cartis, and J. Tanner, "Compressed sensing: How sharp is the restricted isometry property?", *SIAM Rev.*, vol. 53, no. 1, pp. 105-125, 2011.

[16] D. W. Bliss, and K.W. Forsythe, "Multiple-input multiple-output (MIMO) radar and imaging: Degrees of freedom and resolution", *Proc. 37th IEEE Asilomar Conf. Signlas, Systems, Computers,* Nov. 2003, vol. 1, pp. 54-59.

[17] Johann F. Bhme, "Estimation of spectral parameters of correlated signals in wavefields", *Signal Processing* vol. 11, no. 4, pp. 329-337, 1986.

[18] Stephen Boyd, and Lieven Vandenberghe, Convex optimization. *Cambridge University Press*, 2004.

[19] Y. Bresler, M. Gastpar, and R. Venkataramani, "Image compression on-the-fly by universal sampling in Fourier imaging systems", *Proc. DECI*, Santa Fe, NM, pp. 48-48, Feb. 1999.

[20] Y. Bresler and A. Macovski, "On the number of signals resolvable by a uniform linear array," *IEEE Transactions on Acoustics, Speech and Signal Processing* vol. 34, no. 6, pp. 1361- 1375, Dec. 1986.

[21] Homer P. Bucker, "Cross-sensor beam forming with a sparse line array," *J. Acoust. Soc. Am.,* Vol. 61, Issue 2, pp. 494-498, Feb. 1977

[22] Homer P. Bucker, "High resolution cross-sensor beamforming for a billboard array," *J. Acoust. Soc. Am.,* Vol. 65, Issue 1, pp. 145-147, Jan. 1979.

[23] T. T. Cai, L. Wang, and G. Xu, "Stable recovery of sparse signals and an oracle inequality",*IEEE Transactions on Information Theory*, vol. 56, no. 7, pp. 3516-3522, Jul. 2010.

[24] E. J. Candes, "Compressive sampling",*Proceedings of the International Congress of Mathematicians*, pp. 1433-1452, 2006.

[25] E. Candes,"The restricted isometry property and its implications for compressed sensing", *Comptes Rendus Mathematique*, 2008.

[26] E. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information", *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489-509, Feb. 2004.

[27] E. Candes, J. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements", *Communications on Pure and Applied Mathematics*, vol. 59, no. 8, pp. 1207-1223, August 2006.

[28] E. J. Candes and T. Tao, "Decoding by linear programming", *IEEE Transactions on Information Theory*, vol. 51, no. 12, pp. 4203-4215, 2005.

[29] E. J. Candes and T. Tao. "Near-optimal signal recovery from random projections: universal encoding strategies", *IEEE Transactions on Information Theory*, vol. 52, no. 12, pp. 5406-5425, 2006.

[30] J. Capon, "High-resolution frequency-wavenumber spectrum analysis." *Proceedings of the IEEE* vol. 57, no. 8, pp. 1408-1418, 1969.

[31] S. C. Chan, and H. H. Chen, "Uniform Concentric Circular Arrays With Frequency-Invariant Characteristics-Theory, Design, Adaptive Beamforming and DOA Estimation," *IEEE Trans. on Signal Processing*, vol. 55, Issue 1, pp. 165-177, Jan. 2006.

[32] N. R. Chapman and I. Barrodale, "Deconvolution of marine seismic data using the l_1 norm", em Geophysics Journal of the Royal Astronomical Society, vol. 72, pp. 93-100, 1983.

[33] Yih-Min Chen, "On spatial smoothing for two-dimensional direction-of-arrival estimation of coherent signals," Signal Processing, IEEE Transactions on , vol.45, no.7, pp.1689-1696, Jul 1997.

[34] S. Chen and D. Donoho "Basis pursuit",*Twenty-Eighth Asilomar Conference on Signals, Systems and Computers*, 1994.

[35] S. Chen, D. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit", *SIAM Review*, vol. 43, no. 1, pp. 129-159, 2001.

[36] J. Chen and X. Huo, "Theoretical results on sparse representations of multiple-measurement vectors", *IEEE Transactions on Signal Processing*, vol. 54, no. 12, pp. 4634-4643, 2006.

[37] T. Chen, and P. P. Vaidyanathan, "The role of integer matrices in multidimensional multirate systems," *IEEE Trans. Signal Proc.*, vol. SP-41, pp. 1035–1047, May 1993.

[38] Chun-Yang Chen, and P. P. Vaidyanathan, "MIMO Radar Space-time Adaptive Processing Using Prolate Spheroidal Wave Functions," *IEEE Transactions on Signal Processing*, vol. 56, no. 2, pp. 623-635, Feb. 2008.

[39] Chun-Yang Chen, and P. P. Vaidyanathan, "Minimum redundancy MIMO radars," *IEEE International Symposium on Circuits and Systems (ISCAS), 2008*, pp. 45-48, 18-21 May 2008.

[40] Chun-Yang Chen, and P. P. Vaidyanathan. "Compressed sensing in MIMO radar", *42nd Asilomar Conference on Signals, Systems and Computers*, 2008.

[41] Weiguo Chen, Kon M. Wong, and James P. Reilly, "Detection of the number of signals - A predicted eigen-threshold approach", *IEEE Transactions on Signal Processing*, vol. 39, pp. 1088-1098, May 1991.

[42] P. Chevalier, L. Albera, A. Ferreol, P. Comon, "On the virtual array concept for higher order array processing," *IEEE Transactions on Signal Processing*, vol. 53, no. 4, pp. 1254-1271, April 2005.

[43] P. Chevalier, A. Ferreol, "On the virtual array concept for the fourth-order direction finding problem," *IEEE Transactions on Signal Processing*, vol. 47, no. 9, pp. 2592-2595, Sept. 1999.

[44] P. Chevalier, A. Ferreol, and L. Albera, "High-resolution direction finding from higher order statistics: The $2q$-MUSIC algorithm," *IEEE Trans. Signal Process.*, vol. 54, no. 8, pp. 2986-2997, Aug. 2006.

[45] J. Claerbout and F. Muir, "Robust modeling with erratic data", *Geophysics*, vol. 38, pp. 826-844, 1973.

[46] E. Di Claudio, and R. Parisi, "WAVES: weighted average of signal subspaces for robust wideband direction finding," *IEEE Transactions on Signal Processing*, vol.49, no.10, pp.2179-2191, Oct 2001.

[47] A. Cohen, D. Wolfgang, and R. DeVore, "Compressed sensing and best k-term approximation", *Journal of the American Mathematical Society*, vol. 22, no. 1, pp. 211-231, 2009.

[48] S. F. Cotter and B. D. Rao, "Sparse channel estimation via matching pursuit with application to equalization", *IEEE Trans. on Communications*, vol. 50, pp. 374-377, 2002.

[49] S. F. Cotter, B. D. Rao, K. Engan, and K. Kreutz-Delgado, "Sparse solutions to linear inverse problems with multiple measurement vectors",*IEEE Transactions on Signal Processing*, vol. 53, no. 7, pp. 2477-2488, 2005.

[50] A. V. Dandawate and G. B. Giannakis, "Asymptotic theory of mixed time averages and Kth-order cyclic-moment and cumulant statistics," *IEEE Trans. Inf. Theory*, vol. 41, no. 1, pp. 216-232, Jan. 1995.

[51] M. Davenport, M. Wakin, and R. Baraniuk, "Detection and estimation with compressive measurements", *Technical report*, Rice ECE Department, November 2006.

[52] G. Davis, S. Mallat, and M. Avellaneda, "Adaptive greedy approximations", *Journal of Constructive Approximation*, vol. 13, pp. 57-98, 1997.

[53] M. E. Davies, Y. Eldar, "Rank Awareness in Joint Sparse Recovery", *IEEE Transactions on Information Theory*, vol. 58, no. 2, pp. 1135-1146, Feb. 2012.

[54] J. P. Delmas, "Closed-Form Expressions of the Exact Cramer-Rao Bound for Parameter Estimation of BPSK, MSK, or QPSK Waveforms," *IEEE Signal Processing Letters,* vol. 15, pp. 405-408, 2008.

[55] J. P. Delmas and H. Abeida, "Cramer-Rao bounds of DOA estimates for BPSK and QPSK Modulated signals,"*IEEE Transactions on Signal Processing,* vol. 54, no. 1, pp. 117- 126, Jan. 2006.

[56] D. Donoho, "Compressed sensing", *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp.1289-1306, 2006.

[57] D. Donoho, "High-dimensional centrally symmetric polytopes with neighborliness proportional to dimension", *Discrete and Computational Geometry*, vol. 102, no. 27, pp. 617-652, 2006.

[58] D. L. Donoho, and Michael Elad."Optimally sparse representation in general (nonorthogonal) dictionaries via 1 minimization", *Proceedings of the National Academy of Sciences* vol. 100, no. 5 pp. 2197-2202, 2003.

[59] D. Donoho, M. Elad, and V.N. Temlyakov, "Stable recovery of sparse overcomplete representations in the presence of noise", *IEEE Transactions on Information Theory*, vol. 52, no. 1, pp. 6-18, 2006.

[60] D. L. Donoho and X. Huo, "Uncertainty principles and ideal atomic decomposition", *IEEE Transactions on Information Theory*, vol. 47, no. 7, pp. 2845-2862, 2001.

[61] D. Donoho and J. Tanner, "Neighborlyness of randomly-projected simplices in high dimensions", *Proceedings of the National Academy of Sciences*, vol. 102, no. 27, pp. 9452-9457, 2005.

[62] D. Donoho and J. Tanner, "Sparse nonnegative solution of underdetermined linear equations by linear programming", *Proceedings of the National Academy of Sciences*, vol. 102, no. 27, pp. 9446-9451, 2005.

[63] M. C. Dogan, J. M. Mendel, "Applications of cumulants to array processing .I. Aperture extension and array calibration," *IEEE Transactions on Signal Processing,* vol. 43, no. 5, pp. 1200-1216, May 1995.

[64] M. A. Doron and A. J. Weiss, "On focusing matrices for wide-band array processing," *IEEE Transactions on Signal Processing*, vol. 40, issue 6, pp. 1295-1302, Jun 1992.

[65] M. F. Duarte, R. G. Baraniuk, "Kronecker Compressive Sensing", *IEEE Transactions on Image Processing*, vol. 21, no. 2, pp. 494-504, Feb. 2012.

[66] M. F. Duarte, and R. G. Baraniuk. "Spectral compressive sensing", *Applied and Computational Harmonic Analysis*, 2012.

[67] M. Duarte, M. Davenport, D. Takhar, J. Laska, T. Sun, K. Kelly, and R. G. Baraniuk, "Single-pixel imaging via compressive sampling", *IEEE Signal Processing Magazine*, vol. 25, pp. 83-91, 2008.

[68] M. F. Duarte, Y. C. Eldar, "Structured Compressed Sensing: From Theory to Applications", *IEEE Transactions on Signal Processing*, vol. 59, no. 9, pp. 4053-4085, Sept. 2011.

[69] D. E. Dudgeon and R. M. Mersereau, "Multidimensional digital signal processing," Prentice Hall, Inc., Englewoods Cliffs, N.J., 1984.

[70] Y. C. Eldar, P. Kuppinger, and H. Bolcskei, "Block-sparse signals: uncertainty relations and efficient recovery", *IEEE Trans. on Signal Processing*, vol. 58, no. 6, pp. 3042-3054, 2010.

[71] Y. C. Eldar and H. Rauhut "Average case analysis of multichannel sparse recovery using convex relaxation", *IEEE Transactions on Information Theory*, vol. 56, no. 1, pp. 505-519, 2010.

[72] B. L. Evans, "Designing commutative cascades of multidimensional upsamplers and downsamplers," IEEE Signal Processing Letters, vol. 4, no. 11, pp. 313-316, Nov. 1997.

[73] B. L. Evans, R. H. Bamberger, and J. H. McClellan, "Rules for multidimensional multirate structures," IEEE Trans. Signal Processing, vol. 41, pp. 762-771, Apr. 1994.

[74] E. Evans, J. R. Johnson, and D. F. Sun, "High resolution angular spectrum estimation techniques for terrain scattering analysis and angle of arrival estimation," *Proc. 1st ASSP Workshop Spectral Estimation*, pp. 134-139, Hamilton, Ontario, Canada, August 1981.

[75] P. Feng and Y. Bresler, "Spectrum-blind minimum-rate sampling and reconstruction of multiband signals", *Proc. ICASSP*, vol. 3, pp. 16881691, Atlanta, GA, May 1996.

[76] P. Feng, "Universal minimum-rate sampling and spectrum-blind reconstruction for multi-band signals, *Ph.D. dissertation, Univ. Illinois Urbana-Champaign*, Urbana, IL, Dec. 1997.

[77] S. Foucart and M. Lai, "Sparsest solutions of underdetermined linear systems via l_q -minimization for $0 < q < 1$, *Appl. Comput. Harmonic Analysis*, vol. 26, no. 3, pp. 395-407, 2009.

[78] B. Friedlander, A. J. Weiss, "Direction finding using spatial smoothing with interpolated arrays ," *IEEE Transactions on Aerospace and Electronic Systems,* vol. 28, no. 2, pp. 574-587, April 1992.

[79] J. J. Fuchs, "Recovery of exact sparse representations in the presence of noise", *IEEE Transactions on Information Theory*, vol. 51, no. 10, pp. 3601-3608, Oct. 2005.

[80] F. R. Gantmacher, *The theory of matrices,* vol. 1,2, Chelsa Publishing Co., N.Y., 1959.

[81] A.B. Gershman, P. Stoica, M. Pesavento, and E. G. Larsson, "Stochastic Cramer-Rao bound for direction estimation in unknown noise fields," *IEE Proceedings - Radar, Sonar and Navigation*, vol. 149, no. 1, pp. 2-8, Feb. 2002.

[82] M. Ghavami, "Wideband smart antenna theory using rectangular array structures," *Proc. IEEE Trans. on Signal Processing,* vol. 50, Issue 9, pp. 2143-2151, Sept. 2002.

[83] I. Gorodnitsky and B. Rao, "Sparse signal reconstruction from limited data using FOCUSS: a re-weighted norm minimization algorithm", *IEEE Transactions on Signal Processing*, vol. 45, no. 3, pp. 600-616, 1997.

[84] D. Guo, "Neighbor discovery in ad hoc networks as a compressed sensing problem", *Information Theory and Application Workshop*, UCSD, 2009.

[85] G. Harikumar and Y. Bresler, "A new algorithm for computing sparse solutions to linear inverse problems", *Proc. ICASSP*, Atlanta, GA, May 1996, vol. 3, pp. 1331-1334.

[86] J. Haupt, W.U. Bajwa, G. Raz, R. Nowak, "Toeplitz Compressed Sensing Matrices With Applications to Sparse Channel Estimation", *IEEE Transactions on Information Theory*, vol. 56, no. 11, pp.5862-5875, Nov. 2010.

[87] M. A. Herman and T. Strohmer, "High-resolution radar via compressed sensing", *IEEE Transactions on Signal Processing*, vol. 57, no. 6, pp. 2275-2284, 2009.

[88] G. E. Hinton and R. R. Salakhutdinov, "Reducing the Dimensionality of Data with Neural Networks", *Science*, vol. 313, no. 5786, pp. 504-507, July 2005.

[89] R.T. Hoctor, S. A. Kassam, "The unifying role of the coarray in aperture synthesis for coherent and incoherent imaging," *Proceedings of the IEEE,* vol. 78, no. 4, pp. 735-752, April 1990.

[90] B. Hochwald and A. Nehorai, "Identifiability in array processing models with vector-sensor applications," *IEEE Transactions on Signal Processing,* vol. 44, no. 1, pp. 83-95, Jan. 1996.

[91] R. A. Horn and C. R. Johnson, *Matrix analysis,* Cambridge University Press, Cambridge, 1985.

[92] R. A. Horn, and C. R. Johnson. *Topics in Matrix Analysis*. Cambridge University Press, New York: 1991.

[93] H. Hung and M. Kaveh, "Focussing matrices for coherent signal-subspace processing," *IEEE Transactions on Acoustics, Speech and Signal Processing,* vol.36, no.8, pp.1272-1281, Aug 1988.

[94] M. M. Hyder and K. Mahata, "A robust algorithm for joint-sparse recovery",*IEEE Signal Processing Letters*, vol. 16, no. 12, pp. 1091-1094, 2009.

[95] S. Ji, Y. Xue, and L. Carin, "Bayesian compressive sensing", *IEEE Transactions on Signal Processing*, vol. 56, no. 6, pp. 2346-2356, 2008.

[96] Don H. Johnson and Dan E. Dudgeon, *Array Signal Processing - Concepts and Techniques,* Prentice-Hall, Englewood Cliffs, NJ, 1993.

[97] A. A. C. M. Kalker, "Commutativity of up/down sampling," *Electronics letters,* vol. 28, no. 6, , pp. 567-569, March 1992.

[98] M. Kaveh, A. Barabell, "The statistical performance of the MUSIC and the minimum-norm algorithms in resolving plane waves in noise," *IEEE Transactions on Acoustics, Speech and Signal Processing,* vol.34, no.2, pp. 331-341, April 1986.

[99] A. Khajehnejad, W. Xu, S. Avestimehr, and B. Hassibi, "Analyzing weighted l_1 minimization for sparse recovery with nonuniform sparse models",*IEEE Transactions on Signal Processing*, vol. 59, no. 5, pp. 1985-2001, 2011.

[100] C. G. Khatri and C. R. Rao, "Solutions to some functional equations and their applications to characterization of probability distributions",*The Indian J. Stat. Series A*, vol. 30, no. 2, pp. 167-180, 1968.

[101] D. E. Knuth, *The art of computer programming*, vol. 2, Addison Wesley, 1998.

[102] J. Kovačević and M. Vetterli, "The commutativity of up/downsampling in two dimensions," *IEEE Trans. on Info. Theory,* vol. IT-37, pp. 695-698, May 1991.

[103] R. J. Kozick, S. A. Kassam, "Coarray synthesis with circular and elliptical boundary arrays," *IEEE Transactions on Image Processing,* vol. 1, no. 3, pp. 391-405, July 1992.

[104] H. Krim and M. Viberg, "Two decades of array signal processing research: the parametric approach," *IEEE Signal Process. Mag.,* vol. 13, pp. 67-94, Jul. 1996.

[105] J.B. Kruskal, "Three-way arrays: Rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics", *Linear Algebra Applications*, vol. 18, 1977.

[106] T. S. Lee, "Efficient wideband source localization using beamforming invariance technique," *IEEE Transactions on Signal Processing,* vol.42, no.6, pp.1376-1387, Jun 1994.

[107] Kiryung Lee, Y. Bresler, and M. Junge, "Subspace Methods for Joint Sparse Recovery," *IEEE Transactions on Information Theory*, vol. 58, no. 6, pp. 3613-3641, June 2012.

[108] V. P. Leonov, and A. N. Shiryaev, "On a method of calculation of semi-invariants," *Theory Prob. Appl.*, vol. 4, pp. 319-328, 1959.

[109] S. Levy and P. Fullagar, "Reconstruction of a sparse spike train from a portion of its spectrum and application to high-resolution deconvolution", *Geophysics*, vol. 46, pp. 1235-1243, 1981.

[110] D. A. Linebarger, I. H. Sudborough, I. G. Tollis, "Difference bases and sparse sensor arrays," *IEEE Transactions on Information Theory,* vol. 39, no. 2, pp. 716-721, March 1993.

[111] W. Liu, "Adaptive Broadband Beamforming with Spatial-only Information," *Proc. of the 2007 International Conference on Digital Signal Processing (DSP),* pp. 575-578, UK.

[112] W. Liu, D. McLernon, and M. Ghogho, "Frequency Invariant Beamforming Without Tapped Delay-Lines," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2007,* vol. 2, pp. II-997 - II-1000, Apr. 2007.

[113] X. Liu, and N.D. Sidiropoulos, "Almost sure identifiability of constant modulus multidimensional harmonic retrieval," Signal Processing, IEEE Transactions on , vol.50, no.9, pp. 2366- 2368, Sep. 2002.

[114] W. Liu and S. Weiss,"Broadband Beamspace Adaptive Beamforming with Spatial-only Information," *Proc. of the 2008 IEEE Sensor Array and Multi-Channel Signal Processing Workshop,* pp. 330-334, Germany.

[115] W. Liu, R. Wu, and R. J. Langley, "Design and Analysis of Broadband Beamspace Adaptive Arrays," *IEEE Trans. on Antennas and Propagation,* vol. 55, Issue 12, pp. 3413-3420, Dec. 2007.

[116] Z. Lei and A. Makur, "Enumeration of downsampling lattices in two-dimensional multirate systems," IEEE Trans. Signal Processing, vol. 56, pp. 414-418, Jan. 2008.

[117] M. Lustig, D. Donoho, J.M. Santos, and J.M. Pauly, "Compressed sensing MRI", *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 72-82, 2008.

[118] W. -K. Ma, T. -H. Hsieh, and C. -Y. Chi, "DOA estimation of quasi-stationary signals via Khatri-Rao subspace,", *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2009*, pp. 2165-2168, Apr. 2009.

[119] C. C. MacDuffee, *The theory of matrices,* Chelsa Publ. Co., N.Y., 1946.

[120] S. Mac Lane and G. Birkhoff, *Algebra,* Mac Millan Co., N.Y., 1967.

[121] D. Malioutov, M. Cetin, and A. Willsky, "A sparse signal reconstruction perspective for source localization with sensor arrays", *IEEE Transactions on Signal Processing*, vol. 53, no. 8, pp. 3010-3022, 2005.

[122] S. G. Mallat and Z. Zhang "Matching pursuits with time-frequency dictionaries",*IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397-3415, 1993.

[123] N. Meinshausen and P. Buhlmann, "High-dimensional graphs and variable selection with the LASSO, *Ann. Stat.*, vol. 34, pp. 1436-1462, 2006.

[124] J.M. Mendel, "Tutorial on higher-order statistics (spectra) in signal processing and system theory: theoretical results and some applications," *Proceedings of the IEEE*, vol. 79, no. 3, pp. 278-305, Mar 1991.

[125] M. Mishali and Y. Eldar, "Blind multiband signal reconstruction: Compressed sensing for analog signal", *IEEE Transactions on Signal Processing*, vol. 57, no. 3, pp. 993-1009, Mar. 2009.

[126] A. Moffet, "Minimum-redundancy linear arrays," *IEEE Transactions on Antennas and Propagation,* vol. 16, no. 2, pp. 172-175, Mar 1968.

[127] T. Nagell, *Introduction to number theory*, John Wiley & Sons, Inc., NY, 1951.

[128] D. Needell and J. A. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples", *Applied and Computational Harmonic Analysis*, vol. 26, pp. 301-321, 2008.

[129] G. Obozinski, B. Taskar, and M. Jordan, "Joint covariate selection and joint subspace selection for multiple classification problems", *Stat. Comput.*, vol. 20, no. 2, pp. 231-252, 2010.

[130] G. Obozinski, M. Wainwright, and M. Jordan, "Support union recovery in high-dimensional multivariate regression, *Ann. Stat.*, vol. 39, no. 1, pp. 1-47, 2011.

[131] Piya Pal, P. P. Vaidyanathan, "Frequency Invariant MVDR Beamforming without filters and Implementation using MIMO radar," *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2009.

[132] Piya Pal and P.P. Vaidyanathan, "A novel autofocusing approach for estimating directions-of-arrival of wideband signals", *43rd Asilomar Conf. on Signals, Systems and Computers*, pp.1663-1667, Nov. 2009.

[133] Piya Pal and P. P. Vaidyanathan, "A novel array structure for directions-of-arrival estimation with increased degrees of freedom", *Proc. ICASSP.*, Dallas, March 2010.

[134] Piya Pal and P. P. Vaidyanathan, "Beamforming using passive nested arrays of sensors", *Proc. IEEE Int. Symp. Circuits and Systems, ISCAS*, Paris, May-June 2010.

[135] Piya Pal and P. P. Vaidyanathan, "Nested arrays: a novel approach to array processing with enhanced degrees of freedom",*IEEE Transactions on Signal Processing*, vol. 58, pp. 4167-4181, August 2010.

[136] Piya Pal and P. P. Vaidyanathan, "Coprime sampling and the MUSIC algorithm," *IEEE DSP Workshop*, Sedona, AZ, January 2011.

[137] Piya Pal and P. P. Vaidyanathan, "Coprimality of Certain Families of Integer Matrices", *IEEE Trans. on Signal Processing*, vol. 59, pp. 1481–1490, April 2011.

[138] Piya Pal and P. P. Vaidyanathan, "Two Dimensional Nested Arrays on Lattices," ICASSP 2011, May 22-27, 2011, Prague.

[139] Piya Pal and P. P. Vaidyanathan, Non-Uniform Linear Arrays for Improved Identifiability in Cumulant Based DOA Estimation, *Proc. 45th Asilomar Conference on Signals, Systems and Computers*, 2011.

[140] Piya Pal and P. P. Vaidyanathan, "Multiple Level Nested Array: An efficient geometry for 2qth order cumulant based array processing", *IEEE Transactions on Signal Processing*, vol.60, no.3, pp.1253-1269, March 2012.

[141] Piya Pal and P. P. Vaidyanathan,"Correlation Aware Techniques for Sparse Support Recovery", *IEEE Statistical Signal Processing Workshop (SSP)*, 2012, pp.53-56, 5-8 Aug. 2012.

[142] Piya Pal and P. P. Vaidyanathan, "Nested Arrays in Two Dimensions, Part I: Geometrical Considerations", *IEEE Trans. on Signal Processing*, vol. 60, no. 9, pp. 4694-4705, Sept. 2012.

[143] Piya Pal and P. P. Vaidyanathan, "Nested Arrays in Two Dimensions, Part II: Application to Two Dimensional Array Processing," *IEEE Trans. on Signal Processing*, vol. 60, no. 9, pp. 4706-4718, Sept. 2012.

[144] Piya Pal and P. P. Vaidyanathan, "On Application of LASSO for Sparse Support Recovery With Imperfect Correlation Awareness, 46th Asilomar Conference on Signals, Systems and Computers, 2012, Nov. 4-7, 2012.

[145] Piya Pal and P. P. Vaidyanathan, "Correlation-Aware Sparse Support Recovery: Gaussian Sources",*to be presented at International Conference on Acoustics, Speech and Signal Processing, (ICASSP),* 2013.

[146] A. Papoulis, *Signal analysis*, McGraw Hill, Inc., 1977.

[147] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition", *in Proc. 27th Annual Asilomar Conference on Signals, Systems, and Computers*, vol. 1, pp. 40-44, Pacific Grove, CA, Nov. 1993,

[148] D. Pearson, S. U. Pillai, Y. Lee, "An algorithm for near-optimal placement of sensor elements," *IEEE Transactions on Information Theory* vol. 36, no. 6, pp. 1280-1284, Nov. 1990.

[149] S. U. Pillai, Y. Bar-Ness, F. Haber, "A new approach to array geometry for improved spatial spectrum estimation," *Proceedings of the IEEE* , vol. 73, no. 10, pp. 1522-1524, Oct. 1985.

[150] S. Pillai, F. Haber, "Statistical analysis of a high resolution spatial spectrum estimator utilizing an augmented covariance matrix," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 35, no. 11, pp. 1517-1523, Nov 1987.

[151] S. U. Pillai, B. H. Kwon, "Forward/backward spatial smoothing techniques for coherent signal identification," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 37, no. 1, pp. 8-15, Jan 1989.

[152] F. V. Pisarenko, "The retrieval of harmonics from a covariance function", *Geophysical Journal of the Royal Astronomical Society* vol. 33, no. 3 pp. 347-366, 1973.

[153] Y. L Polo, Ying Wang, A. Pandharipande, G. Leus, "Compressive wide-band spectrum sensing," *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing, 2009. ICASSP 2009*, pp.2337-2340, April 2009.

[154] B. Porat, B. Friedlander, "Direction finding algorithms based on high-order statistics," *IEEE Transactions on Signal Processing*, vol. 39, no. 9, pp. 2016-2024, Sep 1991.

[155] Hugh C. Pumphrey, "Design of Sparse Arrays in one, two and three dimensions," *J. Acoust. Soc. Amer.*, vol. 93, Issue 3, pp. 1620-1628, March 1993.

[156] H. Rauhut, "Compressive sensing and structured random matrices", *Theor. Foundations Numer. Meth. Sparse Recovery*, vol. 9, pp. 1-92, 2010.

[157] 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, July 1989.

[158] M. Rudelson and R. Vershynin, "On sparse reconstruction from Fourier and Gaussian measurements, *Commun. Pure Appl. Math.*, vol. 61, no. 8, pp. 1025-1045, 2008.

[159] C. S. Ruf, "Numerical annealing of low-redundancy linear arrays," *IEEE Transactions on Antennas and Propagation*, vol. 41, no. 1, pp. 85-90, Jan 1993.

[160] R. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Transactions on Antennas and Propagation*, vol. 34, no. 3, pp. 276-280, Mar 1986.

[161] T. Sekiguchi, and Y. Karasawa, "Wideband beamspace adaptive array utilizing FIR fan filters for multibeam forming," *IEEE Trans. on Signal Processing*, vol. 48, Issue 1, pp. 277-284, Jan. 2000.

[162] S. Shamsunder and G. B. Giannakis, "Modeling of non-Gaussian array data using cumulants: DOA estimation of more sources with less sensors," *Signal Process.*, vol. 30, no. 3, pp. 279-297, Feb. 1993.

[163] Tie-Jun Shan, M. Wax, T. Kailath, "On spatial smoothing for direction-of-arrival estimation of coherent signals," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 33, no. 4, pp. 806-811, Aug 1985.

[164] N. D. Sidiropoulos and R. Bro, "On the uniqueness of multilinear decomposition of N-way arrays", *J. Chemometrics*, vol. 14, no. 3, pp. 229-239, May 2000.

[165] M. Soumekh, "Synthetic aperture radar signal processing" New York: Wiley, 1999.

[166] P. Stoica, P. Babu, J. Li, "SPICE: A Sparse Covariance-Based Estimation Method for Array Processing",*IEEE Transactions on Signal Processing*, vol. 59, no. 2, pp. 629-638, Feb. 2011.

[167] P. Stoica and T. L. Marzetta, "Parameter estimation problems with singular information matrices," *IEEE Transactions on Signal Processing*, vol. 49, no. 1, pp. 87-90, Jan. 2001.

[168] 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.

[169] 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.

[170] C. Studer, and R.G. Baraniuk, "Dictionary learning from sparsely corrupted or compressed signals", Proc. of IEEE International Conf. on Acoustics, Speech and Signal Processing, pp. 3341–3344, 2012.

[171] D. N. Swingler, and J. Krolik, " Source location bias in the coherently focused high-resolution broad-band beamformer," *IEEE Transactions on Acoustics, Speech and Signal processing*, vol.37, no.1, pp.143-145, Jan 1989.

[172] G. Tang and A. Nehorai, "Performance analysis for sparse support recovery", *IEEE Transactions on Information Theory*, vol. 56, no. 3, pp. 1383-1399, 2010.

[173] H. Taylor, S. Banks, and J. McCoy, "Deconvolution with the l_1 norm, *Geophysics*, vol. 44, no. 1, pp. 39-52, 1979.

[174] Zhi Tian, G. B. Giannakis, "Compressed Sensing for Wideband Cognitive Radios", *IEEE International Conference on Acoustics, Speech and Signal Processing, 2007. ICASSP 2007*, vol. 4, pp. 1357-1360, April, 2007.

[175] R. Tibshirani, "Regression shrinkage and selection via the LASSO", *Journal of the Royal Statistical Society*, Series B, vol. 58, no. 1, pp. 267-288, 1996.

[176] M. E. Tipping, "Sparse Bayesian learning and the relevance vector machine", *Journal of Machine Learning Research*, 2001.

[177] J. A. Tropp, "Greed is good: Algorithmic results for sparse approximation", *IEEE Transactions on Information Theory*, vol. 50, no. 10, pp. 2231-2242, 2004.

[178] J. Tropp, "Algorithms for simultaneous sparse approximation. Part II: Convex relaxation," *Signal Processing*, vol. 86, no. 3, pp. 589-602, 2006.

[179] J. A. Tropp, "Just relax: convex programming methods for identifying sparse signals in noise, "IEEE Transactions on Information Theory", vol. 52, no. 3, pp. 1030-1051, March 2006.

[180] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit, *IEEE Transactions on Information Theory*, vol. 53, no. 12, pp. 4655-4666, 2007.

[181] J. Tropp, A. Gilbert, and M. Strauss, "Algorithms for simultaneous sparse approximation. Part I: Greedy pursuit",*Signal Processing*, vol. 86, no. 3, pp. 572-588, 2006.

[182] P. P. Vaidyanathan, *Multirate systems and filter banks*, Prentice Hall, Englewood Cliffs, N.J., 1993.

[183] P. P. Vaidyanathan and Piya Pal, "Sparse coprime sensing with multidimensional lattice arrays", IEEE Digital Signal Processing Workshop, Sedona, AZ, January 2011.

[184] P. P. Vaidyanathan, and Piya Pal, "Sparse Sensing With Co-Prime Samplers and Arrays," *IEEE Transactions on ignal Processing*, vol. 59, no. 2, pp. 573-586, Feb. 2011.

[185] P.P. Vaidyanathan, and P. Pal, "Theory of Sparse Coprime Sensing in Multiple Dimensions," *IEEE Transactions on ignal Processing*, vol. 59, no. 8, pp. 3592-3608, Aug. 2011.

[186] H. L. Van Trees, *Optimum Array Processing: Part IV of Detection, Estimation and Modulation Theory.* New York: Wiley Interscience, 2002.

[187] Barry D. Van Veen, and Kevin M. Buckley, "Beamforming: A versatile approach to spatial filtering." *IEEE ASSP Magazine*, vol. 5, no. 2, pp. 4-24, 1988.

[188] N. Vaswani and W. Lu, "Modified-CS: Modifying compressive sensing for problems with partially known support",*IEEE Transactions on Signal Processing* vol. 58, no. 9, pp. 4595-4607, 2010.

[189] M. Wainwright, "Information-theoretic limits on sparsity recovery in the high-dimensional and noisy setting", *IEEE Transactions on Information Theory*, no. 12, pp. 5728-5741, Dec 2009.

[190] M. Wainwright, "Sharp thresholds for high-dimensional and noisy sparsity recovery using l1-constrained quadratic programming (LASSO)", *IEEE Transactions on Information Theory*, vol. 55, no. 5, pp. 2183-2202, 2009.

[191] H. Wang and M. Kaveh, "Coherent signal-subspace processing for the detection and estimation of angles of arrival of multiple wide-band sources," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol.33, no.4, pp. 823-831, Aug 1985.

[192] D. B. Ward, R. A. Kennedy, and R. C. Williamson, "Theory and design of broadband sensor arryas with frequency invariant far-field beam patterns, " *Journal of Acoust. Soc. Amer.*, vol. 97, no. 2, pp. 1023-1034, Feb. 1995.

[193] D. B. Ward, R. A. Kennedy, and R. C. Williamson, "FIR filter design for frequency invariant beamformers, " *IEEE Signal Processing Letters*, vol. 3, pp. 69-71, Mar. 1996.

[194] D. B. Ward, Z. Ding, and R. A. Kennedy, "Broadband DOA Estimation using frequency invariant beamforming," *IEEE Trans. on Signal Processing*, vol. 46,no. 5, pp. 1463-1469, May 1998.

[195] M. Wax, "Detection and localization of multiple sources in noise with unknown covariance", *IEEE Transactions on Signal Processing*, vol. 40, no. 1, pp. 245-249, 1992.

[196] M. Wax, Tie-Jun Shan, and T. Kailath, "Spatio temporal spectral analysis by eigenstructure methods," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol.32, no.4, pp. 817-827, Aug 1984.

[197] M. Wax and I. Ziskind, "On unique localization of multiple sources by passive sensor arrays," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 37, no. 7, pp. 996-1000, Jul. 1989.

[198] L. Welch, "Lower Bounds on the Maximum Cross Correlation of Signals", *IEEE Transactions on Information Theory*, vol. 20, no. 3, pp. 397-399, May 1974.

[199] D. Wipf and B. D. Rao, "Sparse Bayesian Learning for Basis Selection", *IEEE Transactions on Signal Processing*, vol. 52, no. 8, pp. 2153-2164, August 2004.

[200] D. Wipf and B. D. Rao, "An empirical Bayesian strategy for solving the simultaneous sparse approximation problem", *IEEE Transactions on Signal Processing*, vol. 55, no. 7, pp. 3704-3716, July 2007.

[201] D. Wipf and S. Nagarajan,"A unified Bayesian framework for MEG/EEG source imaging, *NeuroImage*, pp. 947-966, 2008.

[202] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma,"Robust face recogni- tion via sparse representation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 1-18, 2009.

[203] T. Yingming, C. Yen, X. Wang, "Wideband Spectrum Sensing Based on Sub-Nyquist Sampling", *To Appear in IEEE Transactions on Signal Processing*.

[204] Y. S. Yoon, L. M. Kaplan, and J. H. McClellan, "TOPS: new DOA estimator for wideband signals," *IEEE Transactions on Signal Processing*, vol.54, no.6, pp. 1977-1989, June 2006.

[205] P. Zhao and B. Yu, "On model selection consistency of Lasso", *J. Mach. Learning Res.*, vol. 7, pp. 2541-2567, 2006.

[206] J. Zheng, M. Kaveh, "Sparse Spatial Spectral Estimation: A Covariance Fitting Algorithm, Performance and Regularization",*To Appear in IEEE Transactions on Signal Processing*.

[207] Z. Zhang and B. D. Rao, "Sparse Signal Recovery With Temporally Correlated Source Vectors Using Sparse Bayesian Learning", *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 5, pp. 912-926, Sept. 2011.

[208] Z. Zhang and B. D. Rao, "Extension of SBL Algorithms for the Recovery of Block Sparse Signals With Intra-Block Correlation", *IEEE Transactions on Signal Processing*, vol. 61, no. 8, pp. 2009-2015, April, 2013.

[209] M. D. Zoltowski, "On the performance analysis of the MVDR beamformer in the presence of correlated interference," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 36, no. 6, pp. 945-947, June 1988.

[210] Compressed sensing online resources. http://dsp.rice.edu/cs.

[211] Bayesian compressed sensing online resources:. http://people.ee.duke.edu/lcarin/BCS.html.