PONTIFÍCIA UNIVERSIDADE CATÓLICA
DO RIO DE JANEIRO

## Wesley Souza Leite

## Compressed Sensing Algorithms for Direction of Arrival Estimation with Non-Uniform Linear Arrays

**Dissertação de Mestrado**

Dissertation presented to the Programa de Pós-Graduação em Engenharia Elétrica of PUC-Rio in partial fulfillment of the requirements for the degree of Mestre em Engenharia Elétrica.

Advisor: Prof. Rodrigo Caiado de Lamare

Rio de Janeiro
April 2020

PONTIFÍCIA UNIVERSIDADE CATÓLICA
DO RIO DE JANEIRO

**Wesley Souza Leite**

# Compressed Sensing Algorithms for Direction of Arrival Estimation with Non-Uniform Linear Arrays

Dissertation presented to the Programa de Pós-Graduação em Engenharia Elétrica of PUC-Rio in partial fulfillment of the requirements for the degree of Mestre em Engenharia Elétrica. Approved by the Examination Committee.

**Prof. Rodrigo Caiado de Lamare**
Advisor
Departamento de Engenharia Elétrica – PUC-Rio

**Prof. Rosângela Fernandes Coelho**
Instituto Militar de Engenharia – IME

**Prof. Lukas Tobias Nepomuk Landau**
Departamento de Engenharia Elétrica – PUC-Rio

**Prof. Silvio Bernardes Pinto**
Departamento de Engenharia Elétrica – PUC-Rio

Rio de Janeiro, April the 3rd, 2020

**Wesley Souza Leite**

The author graduated in Electrical Engineering from the Federal University of Maranhão, Brazil, 2014.

*To my parents, for their unrestricted love, loyalty and continuing support.*

# Acknowledgments

First of all, I would like to thank my advisor, Professor Rodrigo Caiado de Lamare, for his always accurate and high-quality guidance, setting my sights higher and expanding my skills in order to develop this work seamlessly.

Then, I wish to acknowledge the support and great love of my family: Antonio Leite, Regina Leite, Regilaine Leite and Ana Caldas. My mother, Regina, deserves all the gratitude I can offer. To my father, Antonio, thank you so much for fulfilling me with courage and fearlessness. To my sister, Regilaine, my deepest gratitude for encouraging me. To my girlfriend, Ana Caldas, thank you for the patience and your love, kindly supporting me during challenging periods.

I would like to thank the members of my defense committee: Professor Rosângela Coelho, Professor Lukas Landau, and Professor Silvio Pinto.

Thank you for those I did not mention explicitly (PUC colleagues and PUC professors) and helped me to accomplish this work.

Then, I wish to thank PUC-Rio for the aids granted.

# Abstract

Leite, Wesley Souza; Caiado de Lamare, Rodrigo (Advisor). **Compressed Sensing Algorithms for Direction of Arrival Estimation with Non-Uniform Linear Arrays**. Rio de Janeiro, 2020. 89p. Dissertação de Mestrado – Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro.

The Direction of Arrival (DoA) estimation or Direction Finding (DF) is a relevant topic for research in areas such as radar, sonar, seismology, electronic surveillance, and wireless communications. This thesis devises a new algorithm that combines a stochastic Maximum Likelihood (ML) method with the widely-known Orthogonal Matching Pursuit (OMP) greedy algorithm, commonly used in sparse recovery with Compressive Sensing (CS). Even though ML techniques are known to be optimal in the mean-squared error sense, achieving the Cramér-Rao Lower Bound (CRLB), the tighter lower bound on estimator variance, they demand a significant, sometimes infeasible, amount of computational resources. On the other hand, departing from a sparsified variant of the data acquisition equation, the problem of finding the sparsest solution of underdetermined systems of equations with OMP has been employed successfully to find the DoA estimates, but with many opportunities for improvement in cases of challenging scenarios. For instance, scenarios with electromagnetic (EM) coupling, low signal-to-noise ratio (SNR), and a limited number of available snapshots (time samples). The proposed difference coarray DoA estimator termed List-Based Maximum Likelihood OMP (LBML-OMP) has shown substantial improvements over traditional and modern techniques, such as OMP, Iterative Hard Thresholding (IHT), and Spatial Smoothing Multiple Signal Classification (SS-MUSIC). It uses a list of candidates generated from the OMP solution and decides for the best based on a limited search using the stochastic ML rule. Thus, it does not perform a grid search with the ML estimator, and this justifies its use in practical scenarios. For the sensing of space-time field, classic and modern non-uniform linear arrays are employed, such as 2-nd Order Nested Array (NAQ2), 2-nd Order Super Nested Array (SNAQ2), Minimum Redundancy Array (MRA), Minimum Hole Array (MHA), and Coprime Array (CPA). Additionally, the estimation is performed under the assumption of EM coupling and noise as disturbing side effects. Furthermore, a new model for difference coarray DoA estimation is developed. It accounts for the finite number of snapshots and has shown to increase the estimation accuracy for all the algorithms, not only LBML-OMP, evidencing secondary sources of error for the difference

coarray transformation. To complement the work, a denoising algorithm called Randomized OMP (RandOMP) was applied to successfully increase the estimation accuracy for difference coarray estimators in scenarios with severe noisy conditions. The contributions of this work relate mainly to the development of a new algorithm and a new difference coarray transformation to improve the DoA estimation accuracy with non-uniform linear arrays. Also, it should be noticed the employment of different geometries for the numerical experiments, making evident the impact of the array sensors' positions in the root mean square error (RMSE) curves.

## Keywords

Sensor array systems;     Parameter estimation;     Direction finding; Non-uniform arrays;     Compressive sensing.

# Resumo

Leite, Wesley Souza; Caiado de Lamare, Rodrigo. **Algoritmos de Sensoriamento Compressivo para Estimação de Direção com Arranjos Lineares Não-Uniformes**. Rio de Janeiro, 2020. 89p. Dissertação de Mestrado – Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro.

O problema de estimação de direção (DoA) de chegada é um importante tópico de pesquisa em áreas como radar, sonar, sismologia, vigilância eletrônica e comunicações sem fio. Este trabalho teve como principal resultado o desenvolvimento de um novo algoritmo que combina o método da máxima verossimilhança (ML) estocástica com o algoritmo ganancioso de busca ortogonal (OMP), comumente empregado em recuperação esparsa com Sensoriamento Compressivo (CS). Muito embora técnicas ML sejam consideradas ótimas em termos de erro médio quadrático, atingindo o limitante inferior de Cramér-Rao (CRLB), o menor limitante inferior para a variância do estimador, estas técnicas demandam de modo significante, às vezes inexequível, os recursos computacionais. Por outro lado, a partir de uma variante esparsa da equação de aquisição de dados, o problema de encontrar a solução mais esparsa possível de sistemas de equações subdeterminados com o algoritmo OMP tem sido empregado de modo satisfatório para encontrar as estimativas de direção, porém com muitas oportunidades de melhoria em casos com cenários sujeitos a condições severas. Por exemplo, cenários com acoplamento eletromagnético (EM), baixa razão sinal-ruído (SNR) e um número limitado de amostras temporais disponíveis. O estimador de direção proposto, baseado em coarranjo diferença, denominado OMP com Máxima Verossimilhança Baseado em Lista (LBML-OMP), apresentou uma melhora significativa no processo de estimação em comparação com técnicas tradicionais e modernas, tais como: OMP, Técnica de Limiar Iterativa (IHT) e Classificação de Múltiplos Sinais Espacialmente Suavizados (SS-MUSIC). A técnica proposta utiliza uma lista de candidatos gerada a partir da solução do algoritmo OMP original e decide pelo melhor a partir de uma busca limitada utilizando o estimador ML estocástico, o que justifica seu uso em cenários práticos. Para a amostragem dos sinais no ambiente, arranjos lineares não-uniformes clássicos e modernos foram empregados, tais como Arranjos Aninhados de Segunda Ordem (NAQ2), Arranjos Aninhados de Segunda Ordem Aperfeiçoados (SNAQ2), Arranjos de Redundância Mínima (MRA) e Arranjos Coprimos (CPA). Além disso, a estimação foi realizada considerando-se o efeito do acoplamento EM e ruído. Ainda, um novo modelo para estimação de direção em coarranjo diferença foi

desenvolvido. Este modelo considera o número de amostras temporais finitas (não-assintótico) e mostrou melhora significativa quando do seu emprego no processo de estimação de direção de todos os algoritmos considerados, não apenas o LBML-OMP, evidenciando fontes secundárias de erro no modelo original estabelecido. De forma a complementar o trabalho, um algoritmo de atenuação de ruído chamado OMP aleatorizado (RandOMP) foi utilizado para aumentar a precisão da estimação em cenários com condições de ruído severas. Neste sentido, as contribuições deste trabalho estão relacionadas principalmente ao desenvolvimento de um novo algoritmo e um novo modelo de transformação em coarranjo diferença de modo a melhorar as estimativas de direção das fontes com arranjos lineares não-uniformes. Além disso, enfatiza-se o emprego de diferentes geometrias para as simulações, tornando-se evidente o impacto da posição dos sensores nas curvas de raiz quadrada do erro médio quadrático (RMSE).

## Palavras-chave

Sistemas de arranjos de sensores;    Estimação de parâmetros; Estimação de direção;    Arranjos não-uniformes;    Sensoriamento compressivo.

# Table of contents

# List of figures

# List of tables

# List of Algorithms

## List of Abreviations

**AD/C** - Analog-to-Digital Converters

**CPA** - Coprime Array

**CRLB** - Cramér-Rao Lower Bound

**CS** - Compressed Sensing or Compressive Sensing

**DAQ** - Data Acquisition

**DCTM** - Difference Coarray Transformation Model

**DF** - Direction Finding

**DoA** - Direction of Arrival

**DoF** - Degrees of Freedom

**EDCTM** - Enhanced Difference Coarray Transformation Model

**HPBW** - Half Power Bandwidth

**i.i.d** - Independent and Identically Distributed

**IF** - Intermediate Frequency

**IHT** - Iterative Hard Thresholding

**LB** - List Based

**MHA** - Minimum Hole Array

**ML** - Maximum Likelihood

**MLE** - Maximum Likelihood Estimator/Estimation

**MMV** - Multiple Measurements Vector

**MRA** - Minimum Redundancy Array

**MUSIC** - Multiple Signal Classification

**MVDR** - Minimum Variance Distortionless Response

**NAQn** - nth-Order Nested Array

**NLA** - Non-Uniform Linear Array

**OMP** - Orthogonal Matching Pursuit

**PDF** - Probability Density Function

**PMF** - Probability Mass Function

**PSD** - Positive Semidefinite

**RandOMP**- Randomized Orthogonal Matching Pursuit

**RIC** - Restricted Isometry Constant

**RIP** - Restricted Isometry Property

**RMSE** - Root Mean Squared Error

**SMV** - Single Measurement Vector

**SNAQn** - nth-Order Super Nested Array

**SNR** - Signal-to-Noise Ratio

**SP** - Subspace Pursuit
**SS-MUSIC**- Spatial Smoothing MUSIC

**UDoF** - Uniform Degrees of Freedom

**ULA** - Uniform Linear Array

## List of Symbols

$K$ - the number of nonzero entries of a vector to be recovered

$\mathbb{N}$ - the set of natural numbers

$\mathbb{Z}$ - the set of integers

$\mathbb{C}$ - the set of complex numbers

$\mathbb{A} \setminus \mathbb{B}$ - the difference between sets $\mathbb{A}$ and $\mathbb{B}$

$\mathbb{L}^*$ - the set $\mathbb{L} \setminus \{0\}$

$\overline{\mathbb{L}}$ - the complement of $\mathbb{L}$

$[N]$ - the set of natural numbers excluding the zero element and not exceeding $N$, i.e, $[N] \triangleq \{1, 2, \ldots, N\}$

$n$ - a natural number indicating the physical positioning of a sensor as multiple of $d$ (minimum inter-element spacing), unless otherwise specified. It is referred as *normalized position.*

$\dim(\mathbf{x})$ - the dimension of a vector $\mathbf{x}$

$\operatorname{supp}(\mathbf{x})$ - the support of a vector $\mathbf{x} \in \mathbb{C}^N$ (index set of its non-zero elements)

$|\mathbb{L}|$ - the cardinality of a set $\mathbb{L}$ (number of elements/objects in the set $\mathbb{L}$)

$\Sigma_K$ - set of all $K$-sparse vectors

$\delta_K(\mathbf{A})$ - restricted isometry constant of order $K$ associated to the matrix $\mathbf{A}$

$\delta(\cdot)$ - unit sample sequence or discrete time impulse

$\boldsymbol{\theta}^g$ - the angle grid, in radians, over which the search procedure is performed

$\mathbb{S}$ - the subset of natural numbers denoting the physical placement of sensors, i.e. $\mathbb{S} = \{0, n_1, \ldots, n_{N-1}\}$ for an array with $N$ sensors

$\mathbb{D}$ - the subset of integers denoting the difference coarray associated with the physical array defined by $\mathbb{S}$

$\underset{N \times N}{\boldsymbol{\Psi}}$ - unitary matrix representing a basis for a generic $N$-dimensional space

$\mathbf{x}_{\mathbb{S}}$ - received signal from an array with sensors' locations defined by $\mathbb{S}$

$\mathbf{x}'_{\mathbb{S}}$ - received signal from an array with sensors' locations defined by $\mathbb{S}$ accounting for EM coupling

$\mathbf{x}_{\mathbb{D}}$ - received signal in a difference coarray domain defined by $\mathbb{D}$

$\mathbf{x}'_{\mathbb{D}}$ - received signal in a difference coarray domain defined by $\mathbb{D}$ accounting for EM coupling

$\mathbf{C}_{\mathbf{x}}$ - received signal $\mathbf{x}$ covariance matrix

$\mathbf{C}'_{\mathbf{x}}$ - received signal $\mathbf{x}$ covariance matrix accounting for EM coupling

$\hat{\mathbf{C}}_{\mathbf{x}}$ - sample received signal $\mathbf{x}$ covariance matrix estimate

$\hat{\mathbf{C}}'_{\mathbf{x}}$ - sample received signal $\mathbf{x}$ covariance matrix estimate accounting for EM coupling

$\mathbf{C}_{\mathbf{s}}$ - source signal $\mathbf{s}$ covariance matrix

$\hat{\mathbf{C}}_{\mathbf{s}}$ - sample source signal $\mathbf{s}$ covariance matrix estimate

$\hat{\mathbf{C}}_{\mathbf{s},ml}$ - source signal $\mathbf{s}$ maximum likelihood covariance matrix estimate

$\dot{\mathbf{q}}$ - the statistical realization of a random vector $\mathbf{q}$

$\mathbf{A}_{\mathbb{L}}(\boldsymbol{\theta})$ - array manifold matrix defined over the angle elements in $\boldsymbol{\theta}$ and the sensors positions given by the set $\mathbb{L}$, which is usually assumed to be either $\mathbb{S}$ or $\mathbb{D}$.

$\mathbf{a}_{\mathbb{L}}(\theta_i)$ - steering vector defined for the angle $\theta_i$ and the sensors' positions given by the set $\mathbb{L}$, which is usually assumed to be either $\mathbb{S}$ or $\mathbb{D}$

$[\mathbf{A}]_{i,j}$ - $(i,j)$-th element of matrix $\mathbf{A}$

$[\mathbf{a}]_i$ - $i$-th element of vector $\mathbf{a}$

$[\mathbb{L}]_i$ - $i$-th object of the totally ordered set $\mathbb{L}$

$\langle \mathbf{s} \rangle_{\mathbb{L}}$ - the vector in $\mathbb{C}^{|\mathbb{L}|}$ which is a restriction of $\mathbf{s}$ to the entries indexed by $\mathbb{L}$

$\langle \mathbf{A} \rangle_{\mathbb{L}}$ - the matrix in $\mathbb{C}^{m \times |\mathbb{L}|}$ which is a restriction of $\mathbf{A} \in \mathbb{C}^{m \times N}$ to the columns indexed by $\mathbb{L}$

$\langle\langle \mathbf{A} \rangle\rangle_{\mathbb{L}}$ - the matrix in $\mathbb{C}^{|\mathbb{L}| \times N}$ which is a restriction of $\mathbf{A} \in \mathbb{C}^{m \times N}$ to the rows indexed by $\mathbb{L}$

$L_K(\cdot)$ - operator that extracts the index set of the $K$ largest magnitude entries of a vector

$L'_K(\cdot)$ - operator that extracts the index set of the $K$ local maxima of a vector

$H_K(\cdot)$ - $K$-th order hard thresholding operator. It sets all but the K largest magnitude entries of a vector to zero.

$H'_K(\cdot)$ - modified K-th order hard thresholding operator. It sets all but the $K$ local maxima of a vector to zero

$\text{vec}(\mathbf{A})$ - operator that produces a column vector in $\mathbb{C}^{mN}$ by stacking the $N$ columns of the matrix $\mathbf{A} \in \mathbb{C}^{m \times N}$ below one another

$\text{vecd}(\mathbf{A})$ - diagonal extraction operator. Produces a column vector in $\mathbb{C}^N$ which consists of the diagonal elements of $\mathbf{A} \in \mathbb{C}^{N \times N}$

$\|\cdot\|_p$ - the $l_p$-quasinorm $(p = 0)$ or $l_p$-norm of a vector

$\text{E}\{\cdot\}$ - the expectation operator

$\mathcal{CN}(\boldsymbol{\mu}|\mathbf{C})$ - the complex normal distribution with mean vector $\boldsymbol{\mu}$ and positive semi-definite covariance matrix $\mathbf{C}$

$\mathbf{I}$ - The identity matrix of appropriate dimensions

$\mathbf{AB}$ - standard matrix product between matrices $\mathbf{A}$ and $\mathbf{B}$

$\mathbf{A} \circ \mathbf{B}$ - Column-wise Khatri-Rao product between matrices $\mathbf{A}$ and $\mathbf{B}$

$\mathbf{A} \otimes \mathbf{B}$ - Kronecker product between matrices $\mathbf{A}$ and $\mathbf{B}$

$\mathbf{A}^H$ - the conjugate transpose or Hermitian transpose of a matrix $\mathbf{A} \in \mathbb{C}^{m \times N}$

$\mathbf{A}^*$ - the conjugate of a matrix $\mathbf{A}$

$\mathbf{A}^T$ - the transpose of a matrix $\mathbf{A}$

$\mathbf{A}^\dagger$ - the Moore-Penrose pseudo-inverse associated to the matrix $\mathbf{A}$

$\langle\mathbf{A}\rangle_\mathbb{L}^\dagger$ - the Moore-Penrose pseudo-inverse associated to the matrix $\langle\mathbf{A}\rangle_\mathbb{L} \in \mathbb{C}^{m \times |\mathbb{L}|}$

$\underline{\mathbf{A}}$ - the hollow matrix associated to matrix $\mathbf{A}$. The off-diagonal elements of $\underline{\mathbf{A}}$ and $\mathbf{A}$ are the same while its main diagonal entries are all zero

$\mathbf{P_A}$ - the orthogonal projection matrix onto the spanning of the columns of the full column rank (tall) matrix $\mathbf{A}$

$\mathbf{P_A}^\top$ - the orthogonal projection matrix onto the null space of the full row rank (fat) matrix $\mathbf{A}^H$

$\mathrm{tr}(\mathbf{A})$ - the trace of a matrix $\mathbf{A}$

$\det(\mathbf{A})$ - the determinant of a matrix $\mathbf{A}$

$\mathrm{diag}(\mathbf{x})$ - the diagonal matrix with main diagonal elements equal to the ones in the vector $\mathbf{x}$

$\mathrm{BW_{HPBW}}$ - half-power beamwidth

$\mathrm{BW_{NN}}$ - notch-notch beamwidth

$a^{(i)}, \mathbf{a}^{(i)}$ or $\mathbf{A}^{(i)}$ - the scalar, vector or matrix indexed to the iteration $i$

$\Delta : \mathbb{C}^m \rightarrow \mathbb{C}^N$ - arbitrary sparse recovery algorithm from $\mathbf{y} \in \mathbb{C}^m$ measurements to the recovered sparse vector $\mathbf{x} \in \mathbb{C}^N$

$\max(a,b)$ - function that computes the maximum value contained in the set $\{a,b\} \in \mathbb{R}^2$

$a \bmod b$ - modulo operator. Returns the remainder of the quotient a/b

Notice that blackboard boldface characters ($\mathbb{S}$) are sets; lowercase normal ($a$) and boldface characters ($\mathbf{a}$) indicate scalars, and column vectors, respectively. Uppercase boldface characters represent matrices ($\mathbf{A}$). A detailed explanation for some of the presented symbols is given appropriately along with the text.

From the definitions of Kronecker and column-wise Khatri-Rao products, it should be noticed that, for $\mathbf{A} = [\mathbf{a}_1, \ldots, \mathbf{a}_N] \in \mathbb{C}^{m \times N}$ and $\mathbf{B} = [\mathbf{b}_1, \ldots, \mathbf{b}_K] \in \mathbb{C}^{p \times K}$,

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{11}\mathbf{B} & \cdots & a_{1N}\mathbf{B} \\ \vdots & \ddots & \vdots \\ a_{m1}\mathbf{B} & \cdots & a_{mN}\mathbf{B} \end{bmatrix} \in \mathbb{C}^{mp \times NK}$$

and

$$\mathbf{A} \circ \mathbf{B} = [\mathbf{a}_1 \otimes \mathbf{b}_1 \mid \ldots \mid \mathbf{a}_N \otimes \mathbf{b}_N] \in \mathbb{C}^{mp \times N^2}$$

where the column-wise Khatri-Rao product is defined only for $N = K$.

# 1
# Introduction

Array processing is a relevant topic for research on radar and sonar systems, wireless communications, medical imaging, seismology, radio astronomy, electronic surveillance, and related subjects [1–4]. The powerful tools of array processing have been used to extract information from impinging signals acquired by a set of sensors arranged in a specially designed structure (termed *geometry*) aiming to estimate their unknown parameters [2].

It may appear in different scenarios and contexts: noisy schemes with signals/interferers being modeled as either deterministic or stochastic according to a specific multivariate statistical distribution, with a parameter set to be estimated. In this regard, one can mention the estimation of a source profile according to different quantities, such as direction, polarization, amplitude, frequency, power, and radial velocity [1, 2, 5, 6].

## 1.1
## Problem Statement

The problem consists of performing an in-depth analysis in the space-time field through a predefined technique that is proven to be capable of describing/tracking all the time-varying/invariant parameters of interest regarding the signals traveling around that environment. This work analyzes this general problem by restricting to the case that the directions of the impinging signals are the quantities to be estimated and considered to be time-invariant during all the estimation time window. Moreover, the signals are modeled as stochastic and drawn from white circular complex Gaussian distributions, uncorrelated with each other, and with the measurement noise.

This problem, namely Direction Finding (DF) or Direction of Arrival (DoA) estimation has been a research topic for many years since its beginnings in the early twentieth century, in signal processing. In 1907-09, the first attempts at DoA estimation started using directional characteristics of antenna elements, at that time named *position finding*. Some years later, phased arrays were introduced to perform this task [5].

After the introduction of phased arrays, the data acquisition process has improved significantly, with enhanced signal-to-interference-plus-noise

NLA ● ✕ ✕ ● ● ● ✕ ●
   1       4 5 6    8

ULA ● ● ● ● ● ● ● ●
   1 2 3 4 5 6 7 8
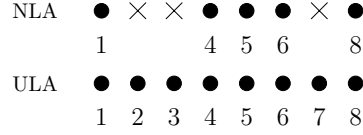
Figure 1.1: Non-Uniform *versus* Uniform Linear Arrays

ratio (SINR) and the parameter estimation task being performed in high dimensional spaces [7].

These arrays consist of a set of sensors arranged according to a specific physical placement rule. Most of the literature evaluate and derive performance prediction equations considering a uniform placement of sensors. The sensors allocation can be carried out in one of three ways: linear (1-D), planar (2-D) or volumetric (3-D) [2]. The present work tackles the 1-D case, with the much broader perspective of considering the non-uniform scenario.

The so-called Non-Uniform Linear Array (NLA) consists of a set of sensors that lacks a regular spatial distribution of the individual elements, i.e., the position difference between any pair of neighbor sensors is not constant. Thus, NLA geometries have some *empty spaces*, which lead to the termed *sparse patterns*.

A schematic that illustrates the differences between uniform and non-uniform arrays is exhibited in Figure 1.1. The filled black circles represent the individual physical sensors, and the crosses mean an empty space in that position. This way of representing non-uniform arrays is useful to compare them with the equivalent ULA of the same aperture. NLA design has attracted considerable attention during the last several decades. Sparse geometries such as Nested Arrays (NAQn), Coprime Arrays (CPA), Minimum Redundancy Arrays (MRA) and Minimum Hole Arrays (MHA or *Golomb rulers*), are typical examples of non-uniform arrays which have been extensively studied recently [8–10].

These non-uniform structures have the advantage that they can more effectively deal with mutual electromagnetic coupling between sensors. Moreover, by denoting $N$ as the number of sensors (physical sensors, unless otherwise specified), these geometries offer $\mathcal{O}(N^2)$ degrees of freedom for DoA estimation because their central difference coarray consecutive set has cardinality $\mathcal{O}(N^2)$, which is called Uniform Degrees of Freedom (UDoF) [1].

Different classes of algorithms have been developed, aiming to estimate the sources DoAs. In this regard, a widely known algorithm category is based upon the maximum likelihood estimation (MLE) technique, which is more accurate, achieving the Cramér-Rao Lower Bound (CRLB), the tighter lower

bound on estimator variance [8, 11]. Therefore, efficient estimators, optimum in the mean-squared error sense, are generated, but at the cost of a significant, sometimes infeasible, amount of computational complexity [2, 12].

On the other hand, some computationally suboptimal feasible alternatives were developed. They reduce computational complexity and speed up the DoA finding procedure. A well-established class in the literature explores the orthogonality between noise and signal subspaces. They are called subspace-based algorithms and include MUSIC, root-MUSIC, and ESPRIT [13, 14]. Other methods called quadratic (deal with quadratic forms in their expressions), such as Minimum Variance Distortionless Response (MVDR - Capon) and Barlett beamformer (beam scan) are also relevant [2, 15, 16]. These techniques give rise to estimators that present a lower computational complexity, but most of the time with an acceptable performance [5].

Additionally, regarding NLA geometries, it is worth emphasizing the recently developed covariance-matrix based method known as SS-MUSIC (Spatial Smoothing MUSIC) [17] to perform MUSIC with a spatially smoothed covariance matrix in coarray domain. Moreover, Compressive Sensing (CS) techniques such as greedy (Orthogonal Matching Pursuit (OMP), Subspace Pursuit (SP)) and thresholding based (Iterative Hard Thresholding (IHT)) are capable of solving the sparsified mathematical models to obtain the DoA estimation for this kind of non-uniform geometry. [18, 19].

Some of the previously mentioned techniques are capable of solving both uniform and non-uniform systems with appropriate mathematical modeling. CS techniques are one of them, with ease of implementation and acceptable computational complexity, as is further discussed.

In this thesis, the main goal is to develop a new CS algorithm capable of better estimating the DoAs associated with multiple sources impinging on an NLA geometry.

## 1.2
## Motivation

Noisy environments and EM coupling are factors that reduce the DoA estimation accuracy [1, 2]. The classic uniform linear geometry has its sensors pretty close to each other, which enhances the EM coupling effect and reduces the aperture size [17]. To overcome this limitation, a lot of non-uniform geometries have been proposed [1, 2, 17, 20]. One of the more significant aspects of these arrangements is that they allow the estimation of more DoAs than sensors (enhanced degrees of freedom) using some specific techniques like SS-MUSIC [17] and CS-based algorithms [19]. In this context, a list-based

CS algorithm was developed with a trade-off between performance and computational complexity. Additionally, the RandOMP denoising method [21] was applied to the DoA estimation task in difference coarray under severe noisy conditions and provided satisfactory results. Furthermore, the traditional CS difference coarray transformation model was slightly modified to account for finite snapshot versions of related quantities.

## 1.3
## Contributions

i) **Greedy CS Algorithm**. A new list-based OMP-like algorithm for DoA estimation in the context of non-uniform arrays is developed. The algorithm takes the OMP suggested solution for the current index support set, searches for $Q$ candidates around it based on the chosen grid, and applies a selection rule for the best candidate, employing a *Maximum Likelihood* search over the candidate set;

ii) **Difference Coarray Transformation Model**. The classic difference coarray model is studied in detail, and a new modeling equation is derived, evidencing some secondary sources of error when performing DoA estimation in difference coarray domain;

iii) **CS-Based Denoising Preprocessing Applied to DoA Estimation with NLA**. The CS-based denoising strategy introduced in [21] was applied to the difference coarray model giving suitable results when acting as a preprocessing to the DoA estimation problem with CS algorithms. This step is incorporated in the proposed framework for DoA estimation as a fundamental first step when dealing with extreme conditions in terms of SNR; and

iv) **Non-Uniform Arrays Performance Evaluation with CS Algorithms** A study of multiple non-uniform linear arrays for DoA estimation under the proposed CS framework with many CS algorithms is performed.

## 1.4
## Notation

See List of Symbols on page 18.

## 1.5
## Thesis Outline

In Chapter 2, the data acquisition model and its modified version to incorporate electromagnetic coupling effects are discussed. Also, a summary of some of the most recent non-uniform geometries is presented. Besides, classical methods for DoA estimation (quadratic, subspace, and maximum likelihood) are summarized. Following that, the recently developed Spatial Smoothing MUSIC (SS-MUSIC) algorithm for DoA estimation in difference coarray domain is described along with some of the CS algorithms used to perform DoA estimation using NLA geometries.

In Chapter 3, a list-based maximum likelihood CS algorithm for DoA estimation is devised and compared with some of the traditional algorithms addressed in Chapter 2. Additionally, a new difference coarray transformation model is developed. Following that, the application of the CS-based denoising algorithm introduced in [21] in the context of non-uniform geometries and difference coarray DoA estimation is proposed.

# 2
# Literature Review and Background

## 2.1
## Introduction

The goal of this chapter is to present a literature revision and a broad overview of the problems commonly faced in array signal processing related to the Direction of Arrival (DoA) estimation. It gives a gradual introduction through the main concepts related to this field of research and defines some of the core technical terms needed to follow the discussion and the approach employed in the text. Firstly, generic data models are defined, including the transformations related to difference coarray domain and electromagnetic coupling model. Secondly, a short review of some classic and modern non-uniform array geometries is performed. After that, it describes some of the traditional techniques to perform DoA estimation with quadratic, subspace-based, and Maximum Likelihood (ML) estimation methods. Lastly, it is presented the algorithms recently developed to perform DoA estimation in difference coarray domain, including the focus of this work, the so-called Compressive Sensing (CS) algorithms. All of the figures presented in this thesis were generated by the author using either MATLAB R2019b (simulations) or TikZ/PGF (LaTeXpackages) (illustrative schemes).

## 2.2
## Data Model

The DoA estimation problem can be performed under different modeling schemes. The choice of a particular model is, in general, related to some aspects, such as desired accuracy, types of signals and noise in the space-time field, array geometry and environment [2]. The parameter estimation is analyzed in the context of Gaussian distributed signals with additive white Gaussian noise. Sources of electromagnetic radiation in far-field regions emit the signals towards the sensor array. Furthermore, the signals are assumed to be narrowband traveling over a isotropic and linear transmission medium [22, 23].

Furthermore, the arrays are supposed to be perfectly calibrated. A

white circular complex Gaussian noise is supposed to be the unique source of perturbation during the data acquisition process. Moreover, the sensors are supposed to have a perfect omnidirectional response and located in positions that are integer multiples of the minimum inter-sensor spacing of $d$.

Figure 2.1 illustrates the DoA estimation scheme: the multiple sources have an expanding wave representation with red concentric partially drawn rings; the array sensors have a 1-D placement distribution over the $x$-axis (linear array) and are deliberately irregularly distributed (non-uniform); there is spatial interference affecting the measurement process (external noise sources possibly not pointlike like natural thermal radiation of the background surrounding the target) [2, 24]; and an array processor performing the DoA estimation from the raw data acquired by the array [5].

Internal noise, although always present (thermal noise, electronic noise), is not represented in the scheme due to the ease of representation, but it is taken into account into the mathematical models.

Associated to the array normal axis there is the broadside axis or $y$-axis. The angle between the source direction from the $k$-th source to the center of the array and the broadside axis is termed broadside angle and is represented by $\theta_k$. That is the quantity of interest. Since the array is illuminated by multiple sources, there is a parameter vector to be estimated that considers the directions $\boldsymbol{\theta}$ estimated as $\hat{\boldsymbol{\theta}}$.

The signals follow the path from the spatially distributed array sensors and enter into an RF network through measurement channels, which are denoted by downward arrows. After that, the preprocessing stage includes a line of frequency converters (tuners - translation to an intermediate frequency (IF)), followed by AD/C converters, down-converters, and a filtering process, which are specially designed to narrowband DoA estimation [5]. Following that, the setting of model parameters is performed to appropriately provide the complete mathematical model for the DoA estimation algorithm.

For this scenario, the single measurement vector (SMV) model for $D$ sources can be modeled by [8, 25, 26]

$$\mathbf{x}_{\mathbb{S}} = \mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})\mathbf{s} + \mathbf{n}_{\mathbb{S}} \tag{2-1}$$

where all the quantities are considered to be in the complex field, except for the DoA vector $\boldsymbol{\theta} \in \mathbb{R}^D$, and the time dependence for the random vectors was not represented. The vector $\mathbf{x}_{\mathbb{S}} \in \mathbb{C}^N$ is the received signal, $\mathbf{s} \in \mathbb{C}^D$ represents the source signal and $\mathbf{n}_{\mathbb{S}} \in \mathbb{C}^N$ is the additive white Gaussian noise.

The matrix $\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}) \in \mathbb{C}^{N \times D}$ is the only deterministic quantity and is called array manifold matrix, defined over the vector $\boldsymbol{\theta}$ and the set $\mathbb{S}$. Its elements
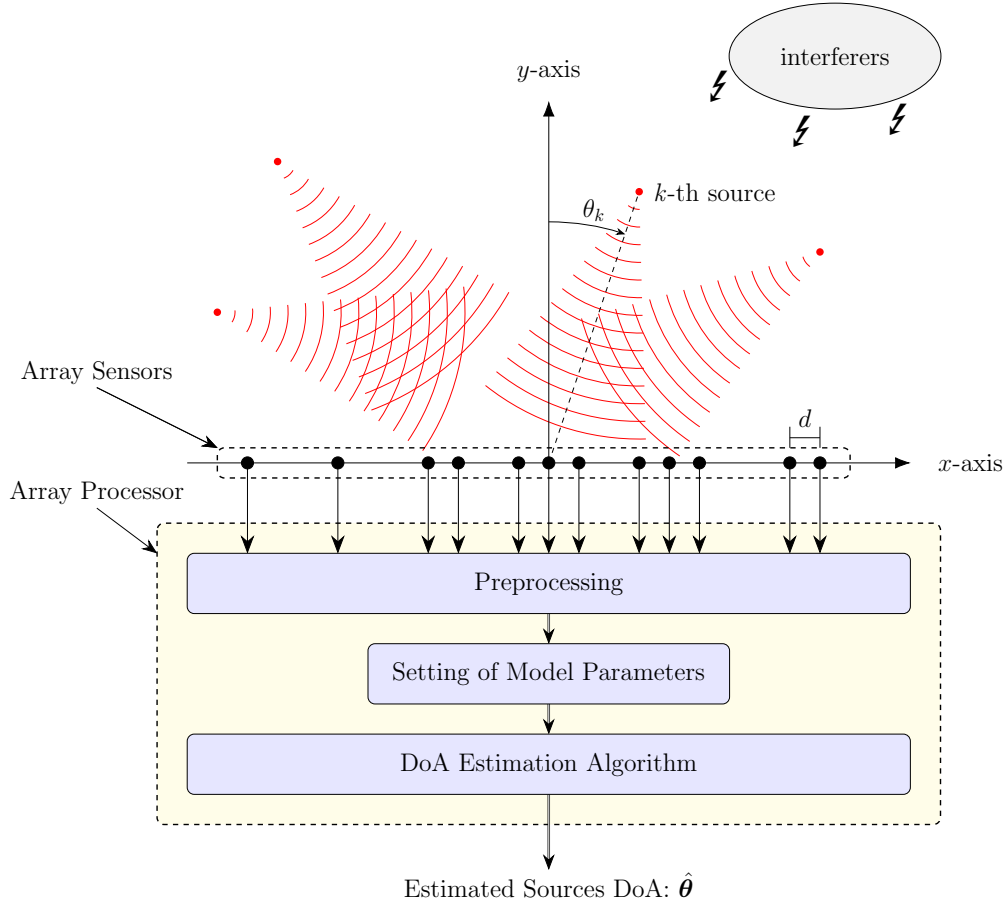
Figure 2.1: DoA Estimation Basic Scheme

are given by

$$[\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})]_{i,k} \triangleq \exp\left(j\frac{2\pi d}{\lambda}n_i \sin(\theta_k)\right) \tag{2-2}$$

where $\theta_k$ are the previously described broadside angles in Figure 2.1, $d$ is the minimum inter-element sensor-to-sensor spacing, $n_i$ are the elements of the set $\mathbb{S}$ (normalized sensors' positions) and $\lambda$ is the source signal carrier wavelength [5].

The set $\mathbb{S} \subset \mathbb{N}$ defines the sensors' physical locations as integer multiples of $d$. Its cardinality $|\mathbb{S}|$ represents the number $N$ of sensors. In order to avoid spatial aliasing, it is assumed $d = \lambda/2$.

The model for DoA estimation can be either deterministic or stochastic [2]. In the following, the latter is adopted. The quantities $\mathbf{s}$ and $\mathbf{n}_{\mathbb{S}}$ are random vectors drawn from a zero-mean multivariate circularly white complex Gaussian distribution with first and second-order statistics given by

$$\mathbf{s} \sim \mathcal{CN}(\mathbf{0}| \operatorname{diag}\left([\sigma_1^2, \ldots, \sigma_D^2]^\top\right))$$
$$\mathbf{n}_\mathbb{S} \sim \mathcal{CN}(\mathbf{0}|\sigma_n^2\mathbf{I})$$

(2-3)

Moreover, considering the partitioned vector $\mathbf{f} = [\mathbf{s}^\top | \mathbf{n}_\mathbb{S}^\top]^\top \in \mathbb{C}^{D+N}$, it should be noticed that, by the following equation

$$\mathbf{f} \sim \mathcal{CN}\left(\left[\begin{array}{c|c} \operatorname{diag}\left([\sigma_1^2, \ldots, \sigma_D^2]^\top\right) & \underset{D \times N}{\mathbf{0}} \\ \hline \underset{N \times D}{\mathbf{0}} & \sigma_n^2\mathbf{I} \end{array}\right]\right)$$

(2-4)

that noise and the source signals are uncorrelated with each other, beyond the fact that the signals are spatially uncorrelated between themselves (diagonal covariance matrix). Since the noise covariance matrix is diagonal, noise from different channels are also uncorrelated. The received signal covariance matrix can be obtained by taking the statistical expectation of the received signal cross-product [2]. Indeed,

$$\begin{aligned}
\mathbf{C}_{\mathbf{x}_\mathbb{S}} &= \mathrm{E}\{\mathbf{x}_\mathbb{S}\mathbf{x}_\mathbb{S}^H\} \\
&= \mathrm{E}\{(\mathbf{A}_\mathbb{S}(\boldsymbol{\theta})\mathbf{s} + \mathbf{n}_\mathbb{S})(\mathbf{A}_\mathbb{S}(\boldsymbol{\theta})\mathbf{s} + \mathbf{n}_\mathbb{S})^H\} \\
&= \mathrm{E}\{(\mathbf{A}_\mathbb{S}(\boldsymbol{\theta})\mathbf{s} + \mathbf{n}_\mathbb{S})\left(\mathbf{s}^H\mathbf{A}_\mathbb{S}^H(\boldsymbol{\theta}) + \mathbf{n}_\mathbb{S}^H\right)\} \\
&= \mathrm{E}\{\mathbf{A}_\mathbb{S}(\boldsymbol{\theta})\mathbf{s}\mathbf{s}^H\mathbf{A}_\mathbb{S}^H(\boldsymbol{\theta})\} + \underbrace{\mathrm{E}\{\mathbf{A}_\mathbb{S}(\boldsymbol{\theta})\mathbf{s}\mathbf{n}_\mathbb{S}^H\}}_{\phantom{x}}{}^{\!\!\!0} + \\
&\quad\quad \underbrace{\mathrm{E}\{\mathbf{n}_\mathbb{S}\mathbf{s}^H\mathbf{A}_\mathbb{S}^H(\boldsymbol{\theta})\}}_{\phantom{x}}{}^{\!\!\!0} + \mathrm{E}\{\mathbf{n}_\mathbb{S}\mathbf{n}_\mathbb{S}^H\} \\
&= \mathbf{A}_\mathbb{S}(\boldsymbol{\theta})\mathrm{E}\{\mathbf{s}\mathbf{s}^H\}\mathbf{A}_\mathbb{S}^H(\boldsymbol{\theta}) + \mathrm{E}\{\mathbf{n}_\mathbb{S}\mathbf{n}_\mathbb{S}^H\} \\
&= \mathbf{A}_\mathbb{S}(\boldsymbol{\theta})\mathbf{C}_\mathbf{s}\mathbf{A}_\mathbb{S}^H(\boldsymbol{\theta}) + \mathbf{C}_{\mathbf{n}_\mathbb{S}} \\
&= \mathbf{A}_\mathbb{S}(\boldsymbol{\theta})\operatorname{diag}\left[\sigma_1^2, \ldots, \sigma_D^2\right]\mathbf{A}_\mathbb{S}^H(\boldsymbol{\theta}) + \sigma_n^2\mathbf{I} \\
&= \sum_{i=1}^{D} \sigma_i^2 \mathbf{a}_\mathbb{S}(\theta_i)\mathbf{a}_\mathbb{S}^H(\theta_i) + \sigma_n^2\mathbf{I}
\end{aligned}$$

(2-5)

where the cross-terms are zero because signal and noise are uncorrelated. This covariance matrix is usually estimated through the sample covariance matrix [27], which gives

$$\hat{\mathbf{C}}_{\mathbf{x}_\mathbb{S}} = \frac{1}{T}\dot{\mathbf{X}}_\mathbb{S}\dot{\mathbf{X}}_\mathbb{S}^H$$

(2-6)

where $T$ is the number of snapshots, $\dot{\mathbf{X}}_\mathbb{S} = [\dot{\mathbf{x}}_\mathbb{S}(1), \dot{\mathbf{x}}_\mathbb{S}(2), \ldots, \dot{\mathbf{x}}_\mathbb{S}(T)] \in \mathbb{C}^{N \times T}$ is the received signal data matrix, and $\dot{\mathbf{x}}_\mathbb{S}(t) \in \mathbb{C}^N$ is the $t$-th statistical realization of the random vector $\mathbf{x}_\mathbb{S}(t)$.

Furthermore, it is worth emphasizing that all the $\dot{\mathbf{x}}_\mathbb{S}(t)$ are drawn from multivariate complex zero-mean Gaussian distributions, and independent and

identically distributed (i.i.d) from snapshot to snapshot[1]. For the sample sources covariance matrix, consider

$$\hat{\mathbf{C}}_{\mathbf{s}} = \frac{1}{T}\dot{\mathbf{S}}\dot{\mathbf{S}}^H \tag{2-7}$$

where $\dot{\mathbf{S}} \in \mathbb{C}^{D\times T}$ is the matrix with source signal snapshots vector as its columns. To conclude, the finite snapshot version of the multiple measurement vector (MMV) model is given by

$$\dot{\mathbf{X}}_{\mathbb{S}} = \mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})\dot{\mathbf{S}} + \dot{\mathbf{N}} \tag{2-8}$$

where $\dot{\mathbf{N}} \in \mathbb{C}^{N\times T}$ is the matrix with the successive snapshots as its columns.

### 2.2.1
### Difference Coarray Transformation Model

In order to deal with the position difference of the array sensors and increase the available Degrees of Freedom (DoF) for DoA estimation, (2-5) is submitted to a transformation termed Difference Coarray Transformation Model (DCTM) by vectorizing the received signal covariance matrix using the $\mathrm{vec}\,(\cdot)$ operator[2] [2, 28] and extract the diagonal elements of the sources covariance matrix with $\mathrm{vecd}\,()$. That said, consider the following

$$
\begin{aligned}
\mathrm{vec}\,(\mathbf{C}_{\mathbf{x}_{\mathbb{S}}}) &= \mathrm{vec}\,\left(\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})\,\mathrm{diag}\left[\sigma_1^2,\ldots,\sigma_D^2\right]\mathbf{A}_{\mathbb{S}}^H(\boldsymbol{\theta}) + \sigma_n^2\mathbf{I}\right) \\
&= \mathrm{vec}\,\left(\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})\,\mathrm{diag}\left[\sigma_1^2,\ldots,\sigma_D^2\right]\mathbf{A}_{\mathbb{S}}^H(\boldsymbol{\theta})\right) + \sigma_n^2\,\mathrm{vec}\,(\mathbf{I}) \\
&= (\mathbf{A}_{\mathbb{S}}^*(\boldsymbol{\theta}) \circ \mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}))\,\mathrm{vecd}\left(\mathrm{diag}\left[\sigma_1^2,\ldots,\sigma_D^2\right]\right) + \sigma_n^2\,\mathrm{vec}\,(\mathbf{I}) \\
&= (\mathbf{A}_{\mathbb{S}}^*(\boldsymbol{\theta}) \circ \mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}))\,\mathbf{p} + \sigma_n^2\tilde{\mathbf{i}}
\end{aligned} \tag{2-9}
$$

where

$$\mathbf{p} = \left[\sigma_1^2,\ldots,\sigma_D^2\right]^\top \tag{2-10}$$

is the sources power/variances vector,

$$\mathbf{A}_{\mathbb{S}}^*(\boldsymbol{\theta}) \circ \mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}) = [\mathbf{a}_{\mathbb{S}}^*(\boldsymbol{\theta}_1) \otimes \mathbf{a}_{\mathbb{S}}(\boldsymbol{\theta}_1),\ldots,\mathbf{a}_{\mathbb{S}}^*(\boldsymbol{\theta}_D) \otimes \mathbf{a}_{\mathbb{S}}(\boldsymbol{\theta}_D)] \tag{2-11}$$

is the column-wise Khratri-Rao product between matrices $\mathbf{A}_{\mathbb{S}}^*(\boldsymbol{\theta})$ and $\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})$, which consists of the Kronecker product of the corresponding columns, and $\tilde{\mathbf{i}} = \mathrm{vec}\,(\mathbf{I}) \in \mathbb{R}^{N^2}$ is the vectorization of the identity matrix.

After these considerations, it is straightforward to notice that

---

[1]Notice that, by (2-1), $\mathbf{x}_{\mathbb{S}}$ comes from a linear transformation of random vectors that does not affect the resulting mean ($\mathbf{s}$ and $\mathbf{n}_{\mathbb{S}}$ are zero-mean), but it does change the covariance matrix.

[2]$\mathrm{vec}\,()$ operator property: $\mathrm{vec}\,\left(\mathbf{A}\mathbf{X}\mathbf{B}^\top\right) = (\mathbf{B} \circ \mathbf{A})\,\mathrm{vecd}\,(\mathbf{X})$ if $\mathbf{X}$ is diagonal. See (5) in [28].

$$[\mathbf{A}_{\mathbb{S}}^*(\boldsymbol{\theta}) \circ \mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})]_{i,k} \triangleq \exp\left(j\frac{2\pi d}{\lambda}(n_{\lfloor(i-1)/N\rfloor} - n_{(i-1)\bmod N})\sin(\theta_k)\right) \quad (2\text{-}12)$$

where $\mathbb{S} = [n_0, n_1, \ldots, n_{N-1}]$ is the set with the sensors normalized positions[3], $\lfloor\cdot\rfloor$ is the *floor* function, and $(\cdot)\bmod(\cdot)$ is the modulo operator (remainder after division).

At this point, it is worth establishing some difference coarray related definitions [1].

**Definition 2.1** *(Difference Coarray Set - $\mathbb{D}$) The difference coarray set represented by $\mathbb{D}$ is a set associated to the sensors' positions $\mathbb{S}$ through*

$$\mathbb{D} \triangleq \left\{ n_1 - n_2 \mid (n_1, n_2) \in \mathbb{S}^2 \right\} \quad (2\text{-}13)$$

*arranged in ascending order and without any repetition of elements.*

**Definition 2.2** *(Degrees of Freedom - DoF) The number of Degrees of Freedom, denoted by DoF, of a geometry specified by $\mathbb{S}$ is the cardinality of its difference coarray set. Then,*

$$\text{DoF} \triangleq |\mathbb{D}| \quad (2\text{-}14)$$

**Definition 2.3** *(Central Difference Coarray Consecutive Set - $\mathbb{U}$) The central difference coarray consecutive set, denoted by $\mathbb{U}$, associated with a difference coarray set $\mathbb{D}$ is the set[4]*

$$\mathbb{U} = \{0\} \cup \{\pm[m]\} \quad (2\text{-}15)$$

*arranged in ascending order and without any repetition of elements, such that $m$ is chosen to maximize the cardinality of $\mathbb{U}$ with $\mathbb{U} \subseteq \mathbb{D}$*

**Definition 2.4** *(Uniform Degrees of Freedom - UDoF) The number of Uniform DoF, denoted by UDoF, of a geometry specified by $\mathbb{S}$, is the cardinality of its central difference coarray consecutive set ($\mathbb{U}$), i.e.,*

$$\text{UDoF} \triangleq |\mathbb{U}| \quad (2\text{-}16)$$

**Definition 2.5** *(Weight Function - $w(m)$) The weight function is an application*

$$w : \mathbb{Z} \to \{0\} \cup \{\pm[N]\}$$

*such that*

$$w(m) = \begin{cases} |\mathbb{M}(m)| & \text{if } m \in \mathbb{D} \\ 0 & \text{otherwise} \end{cases} \quad (2\text{-}17)$$

*where $\mathbb{M}(m) = \{(n_1, n_2) \in \mathbb{S}^2 \mid n_1 - n_2 = m\}$, i.e., $\mathbb{M}(m)$ is a function that counts the number of sensor pairs with separation $m$ (covariance lag).*

---

[3]Usually, we assume $n_0 = 0$ (reference)
[4]The symbol $[m]$ represents the set $\{1, 2, \ldots, m\}$.

Notice that the weight function assumes its maximum value at the origin $(w(0) = N)$, and it possesses even symmetry. The importance of this function is associated with the way it allows one to obtain the quantities DoF, UDoF, and an idea about the array electromagnetic coupling robustness.

Going back to (2-12), it can be seen that $\mathbf{A}_{\mathbb{S}}^*(\boldsymbol{\theta}) \circ \mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})$ will exhibit some repeated rows, once the same position difference $n_{i_2} - n_{i_1}$ can be produced, in general, by more than one pair of sensors, i.e., the perfect array condition [2]

$$w(m) = \begin{cases} 1 & \text{if } m \in \mathbb{D} \setminus \{0\} \\ N & \text{if } m = 0 \\ 0 & \text{otherwise} \end{cases} \tag{2-18}$$

is not fulfilled for the vast majority of the arrays. The notation $\{\mathbb{D}\} \setminus \{0\}$ represents the difference between the sets $\mathbb{D}$ and $\{0\}$. Each sensor pair repetition will give rise to one repeated row in (2-11). If we exclude these rows, we arrive at

$$\begin{aligned} \mathbf{x}_{\mathbb{D}} &= \mathbf{A}_{\mathbb{D}}(\boldsymbol{\theta})\mathbf{p} + \sigma_n^2 \mathbf{i} \\ &= \begin{bmatrix} \mathbf{A}_{\mathbb{D}}(\boldsymbol{\theta}) \mid \mathbf{i} \end{bmatrix} \begin{bmatrix} \mathbf{p} \\ \sigma_n^2 \end{bmatrix} \end{aligned} \tag{2-19}$$

which can be compared to (2-1) and considered to be the deterministic received signal $\mathbf{x}_{\mathbb{D}} \in \mathbb{C}^{\text{DoF}}$ expression with deterministic additive noise $\sigma_n^2 \mathbf{i} \in \mathbb{R}^{\text{DoF}}$ and difference coarray manifold $\mathbf{A}_{\mathbb{D}}(\boldsymbol{\theta}) \in \mathbb{C}^{\text{DoF} \times D}$. In this case, the vector $\mathbf{i}$ has DoF elements. Its $((\text{DoF}+1)/2)$-th entry is unitary and all the others are zero. This can be represented in a compact notation as $[\mathbf{i}]_k = \delta((k - (\text{DoF}+1)/2))$, where $\delta(n)$ is the widely known *unit sample sequence* or *discrete time impulse* [29].

### 2.2.2
### Electromagnetic Coupling Model

One of the most essential features to justify the employment of non-uniform arrays is the increase in robustness to deal with electromagnetic coupling effects [1, 30]. It is straightforward to see that, in (2-1) the sensors measurements do not interfere with each other.

Nevertheless, the mutual electromagnetic coupling effect must be taken into account in order to ensure a higher quality DoA estimation in practice. The model with electromagnetic coupling can be adapted from the previous expressions through

$$\mathbf{x}_{\mathbb{S}}' = \mathbf{G}\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})\mathbf{s} + \mathbf{n}_{\mathbb{S}}, \tag{2-20}$$

where $\mathbf{G} \in \mathbb{C}^{N \times N}$ is the mutual coupling matrix with elements given by

$$[\mathbf{G}]_{i,j} = \begin{cases} g\left(|n_i - n_j|\right) & \text{if } |n_i - n_j| \leq B \\ 0 & \text{otherwise} \end{cases} \tag{2-21}$$

This is called B-banded mode model, and it assumes that the coupling coefficients are inversely proportional to the distance between any two sensors [31, 32]. In that way, it is reasonable to consider the coupling coefficients as zeros for two sensors too far apart. This threshold distance for which there is a minimum coupling defines the parameter $B$.

Moreover, regarding the elements of matrix $\mathbf{G}$, we state that

$$1 = g(0) > |g(m_1)| > |g(m_2)| > \ldots > |g(B)| >$$
$$|g(B+1)| = \ldots = \left|g\left(\max\left(\mathbb{D}^+\right)\right)\right| = 0 \tag{2-22}$$

where it was assumed[5] $m_i \in \mathbb{D}^+$, the sorted strictly positive part of $\mathbb{D}$ and $B < \max\left(\mathbb{D}^+\right)$.

Furthermore, it is assumed that the coupling between two equally spaced sensors is the same and time-invariant during the DoA estimation process. This is not entirely true, once the coupling matrix tends to change with time influenced by temperature, pressure, humidity, and neighboring objects [31].

In this sense, the model in (2-5) for the EM coupling scenario becomes

$$\begin{aligned}
\mathbf{C}'_{\mathbf{x}_{\mathbb{S}}} &= \mathrm{E}\{\mathbf{x}'_{\mathbb{S}}\mathbf{x}'^H_{\mathbb{S}}\} \\
&= \mathrm{E}\{(\mathbf{G}\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})\mathbf{s} + \mathbf{n}_{\mathbb{S}})(\mathbf{G}\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})\mathbf{s} + \mathbf{n}_{\mathbb{S}})^H\} \\
&= \mathrm{E}\{(\mathbf{G}\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})\mathbf{s} + \mathbf{n}_{\mathbb{S}})\left(\mathbf{s}^H\mathbf{A}^H_{\mathbb{S}}(\boldsymbol{\theta})\mathbf{G}^H + \mathbf{n}^H_{\mathbb{S}}\right)\} \\
&= \mathrm{E}\{\mathbf{G}\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})\mathbf{s}\mathbf{s}^H\mathbf{A}^H_{\mathbb{S}}(\boldsymbol{\theta})\mathbf{G}^H\} + \mathrm{E}\{\mathbf{G}\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})\mathbf{s}\mathbf{n}^H_{\mathbb{S}}\}^{\nearrow 0} + \\
&\qquad \mathrm{E}\{\mathbf{n}_{\mathbb{S}}\mathbf{s}^H\mathbf{A}^H_{\mathbb{S}}(\boldsymbol{\theta})\mathbf{G}^H\}^{\nearrow 0} + \mathrm{E}\{\mathbf{n}_{\mathbb{S}}\mathbf{n}^H_{\mathbb{S}}\} \\
&= \mathbf{G}\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})\mathrm{E}\{\mathbf{s}\mathbf{s}^H\}\mathbf{A}^H_{\mathbb{S}}(\boldsymbol{\theta})\mathbf{G}^H + \mathrm{E}\{\mathbf{n}_{\mathbb{S}}\mathbf{n}^H_{\mathbb{S}}\} \\
&= (\mathbf{G}\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}))\,\mathbf{C_s}\,(\mathbf{G}\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}))^H + \mathbf{C_n} \\
&= (\mathbf{G}\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}))\,\mathrm{diag}\left[\sigma_1^2, \ldots, \sigma_D^2\right](\mathbf{G}\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}))^H + \sigma_n^2\mathbf{I} \\
&= \sum_{i=1}^{D} \sigma_i^2 \mathbf{G}\mathbf{a}_{\mathbb{S}}(\theta_i)\mathbf{a}^H_{\mathbb{S}}(\theta_i)\mathbf{G}^H + \sigma_n^2\mathbf{I}
\end{aligned} \tag{2-23}$$

By applying the difference coarray transformation carried out in Subsection 2.2.1, we have[6] [20]

[5]To clarify: $\mathbb{D} = \mathbb{D}^- \cup \{0\} \cup \mathbb{D}^+$, i.e., for $\mathbb{D} = \{-4, -3, -1, 0, 1, 3, 4\} \Rightarrow \mathbb{D}^+ = \{1, 3, 4\}$

[6]See properties (24) $((\mathbf{C} \otimes \mathbf{D})(\mathbf{A} \circ \mathbf{B}) = (\mathbf{C}\mathbf{A}) \circ (\mathbf{D}\mathbf{B}))$ and (5) $(\mathrm{vec}\left(\mathbf{A}\mathbf{X}\mathbf{B}^\top\right) = (\mathbf{B} \circ \mathbf{A})\,\mathrm{vecd}\,(\mathbf{X})$ if $\mathbf{X}$ is diagonal) in [30] and [28], respectively.

$$
\begin{aligned}
\operatorname{vec}\left(\mathbf{C}'_{\mathbf{x}_{\mathbb{S}}}\right) &= \operatorname{vec}\left((\mathbf{G}\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}))\operatorname{diag}\left[\sigma_1^2,\ldots,\sigma_D^2\right](\mathbf{G}\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}))^H + \sigma_n^2\mathbf{I}\right) \\
&= \operatorname{vec}\left((\mathbf{G}\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}))\operatorname{diag}\left[\sigma_1^2,\ldots,\sigma_D^2\right](\mathbf{G}\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}))^H\right) + \sigma_n^2\operatorname{vec}\left(\mathbf{I}\right) \\
&= ((\mathbf{G}\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}))^* \circ (\mathbf{G}\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})))\operatorname{vecd}\left(\operatorname{diag}\left[\sigma_1^2,\ldots,\sigma_D^2\right]\right) + \sigma_n^2\operatorname{vec}\left(\mathbf{I}\right) \\
&= (\mathbf{G}^* \otimes \mathbf{G})(\mathbf{A}_{\mathbb{S}}^*(\boldsymbol{\theta}) \circ \mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}))\,\mathbf{p} + \sigma_n^2\operatorname{vec}\left(\mathbf{I}\right) \\
&= \mathbf{G}'(\mathbf{A}_{\mathbb{S}}^*(\boldsymbol{\theta}) \circ \mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}))\,\mathbf{p} + \sigma_n^2\tilde{\mathbf{i}}
\end{aligned}
$$

$$(2\text{-}24)$$

where $\mathbf{G}' = \mathbf{G}^* \otimes \mathbf{G} \in \mathbb{C}^{N^2 \times N^2}$.

If the repeated rows in $\mathbf{A}_{\mathbb{S}}^*(\boldsymbol{\theta}) \circ \mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})$ are removed (the same procedure performed in (2-19)), then (2-20) becomes

$$
\begin{aligned}
\mathbf{x}'_{\mathbb{D}} &= \mathbf{G}''\mathbf{A}_{\mathbb{D}}(\boldsymbol{\theta})\mathbf{p} + \sigma_n^2\mathbf{i} \\
&= \left[\mathbf{G}''\mathbf{A}_{\mathbb{D}}(\boldsymbol{\theta}) \,\middle|\, \mathbf{i}\right]\begin{bmatrix}\mathbf{p} \\ \sigma_n^2\end{bmatrix}
\end{aligned}
$$

$$(2\text{-}25)$$

which is the model to be used with CS-based DoA estimation algorithms.

Notice that the EM coupling case is a more general scenario. The EM coupling-free setting can be considered in (2-20) by setting $\mathbf{G}$ as the identity matrix, i.e., $B = 0$.

Another relevant concept for EM coupling evaluation in array processing is the so-called coupling leakage coefficient [20], given by

$$
L\left(\mathbf{G}\right) = \frac{\|\underline{\mathbf{G}}\|_F}{\|\mathbf{G}\|_F}
$$

$$(2\text{-}26)$$

where $\underline{\mathbf{G}} = \mathbf{G} - \operatorname{diag}\left(\operatorname{vecd}\left(\mathbf{G}\right)\right)$ is the hollow matrix associated to the matrix $\mathbf{G}$. Usually, arrays that are more robust to EM coupling exhibits a smaller leakage coefficient.

## 2.3
## Array Geometries

In this section, different ways of defining the set $\mathbb{S}$ will be described. New geometries are associated with individual formation rules/tabulated entries. These arrangements will be demonstrated to have different associated characteristics regarding estimation performance, degrees of freedom, leakage coefficient, and electromagnetic coupling. Moreover, since the discussion is restricted to the context of linear arrays, the set $\mathbb{S}$ is the only quantity that defines the geometries.

### 2.3.1
### Uniform Linear Arrays

The so-called Uniform Linear Array (ULA) is the most common array geometry in the literature. Its characteristics have been extensively studied since the very beginning of the array processing theory. The essential characteristics of that kind of array are:

i) Uniformly (equally) spaced physical sensors; and

ii) Linear disposition (over a straight line) of the array elements.

The related literature presents tons of expressions in order to evaluate the performance prediction of different DoA estimation algorithms under a ULA acquisition geometry. However, these expressions become a lot more sophisticated in the context of non-uniform arrays [2].

An example of ULA is given in Figure 2.2 along with its associated triangular-shaped weight function. One must observe that the ULA acronym employed here is the same as *dense ULA*. A sparse ULA is a different arrangement that is detailed in the following sections.
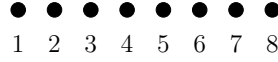


2.2(a): Geometry
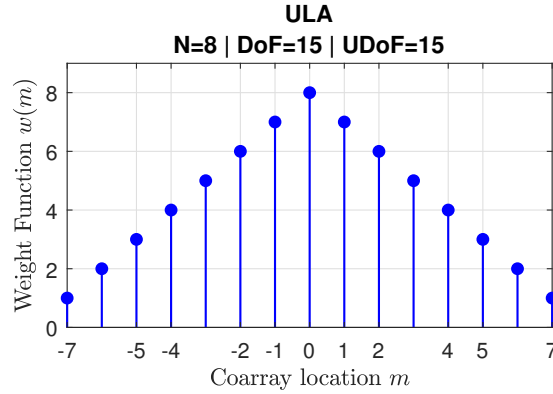


2.2(b): Weight function with UDoF = DoF = 15

Figure 2.2: ULA with $N = 8$ sensors

### 2.3.2
### Non-uniform Linear Arrays

Non-uniform Linear Arrays (NLA) are arrays in which the sensors are linearly distributed with non-uniform spacing [2]. In this text, these structures are treated as having some *empty locations* for some positions in comparison

with a ULA having the same aperture. A generic example is given in Figure 2.3. NLAs have some advantages over ULA, namely:

i) Smaller electromagnetic coupling (EM), once the mean distance between elements is larger than that of a ULA;

ii) Larger aperture with the same number of physical sensors;

iii) Possibility of estimating $\mathcal{O}(N^2)$ sources DoAs with $\mathcal{O}(N)$ physical sensors. With the same number of sensors, ULA arrangements can estimate only $\mathcal{O}(N)$ DoAs.
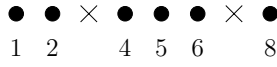


Figure 2.3: Generic NLA with $N = 6$ sensors (filled black circles) and 2 empty locations (crosses)

### 2.3.2.1
### Nested Arrays

The so-called n*th*-Order Nested Arrays (NAQn) were first introduced in [17]. The discussion is restricted here to the 2nd-Order Nested Arrays (NAQ2). The formation rule (positioning procedure) described by $\mathbb{S}$ is given by (2-27). The total number of sensors is $N = N_1 + N_2$. It is straightforward to notice that NAQ2 arrangements consist of two ULA segments. The first one with $N_1$ sensors and unitary spacing and the second with $N_2$ sensors and spacing of magnitude $N_1 + 1$.

$$\mathbb{S} = \{n, \ n = 0, 1, \ldots, N_1 - 1\} \cup \{n(N_1 + 1) - 1, \ n = 1, 2, \ldots, N_2\} \quad (2\text{-}27)$$

That said, the first ULA is a dense structure, and the second one is termed sparse. Two of its fundamental characteristics are commonly highlighted:

i) Filled difference coarray (hole free), i.e.,

$$\mathbb{M}(m) \neq \varnothing \ \ \forall \ \ m \leq \max(\mathbb{D}) \quad (2\text{-}28)$$

ii) DoF and UDoF are obtained through (2-29). From that, notice that there are $\mathcal{O}(N_1 N_2)$ available DoF for arrays with only $N_1 + N_2$ sensors. Besides, since the two kinds of DoF are equal, we state that NAQ2 structures have no holes at all in difference coarray domain, which is an interesting property for covariance matrix based algorithms.

$$\text{DoF} = \text{UDoF} = 2N_2(N_1 + 1) - 1 \quad (2\text{-}29)$$

Figure 2.4 shows a generic array geometry with $N = 8$ sensors (Figure 2.4(a)) assembled with two ULAs formed of 4 sensors each. Figure 2.4(b) represents the associated weight function exhibiting a hole-free difference coarray set, according to what was expected.
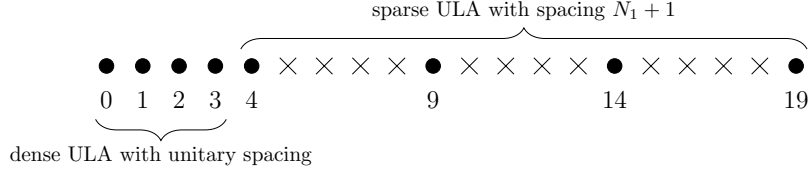


2.4(a): Geometry
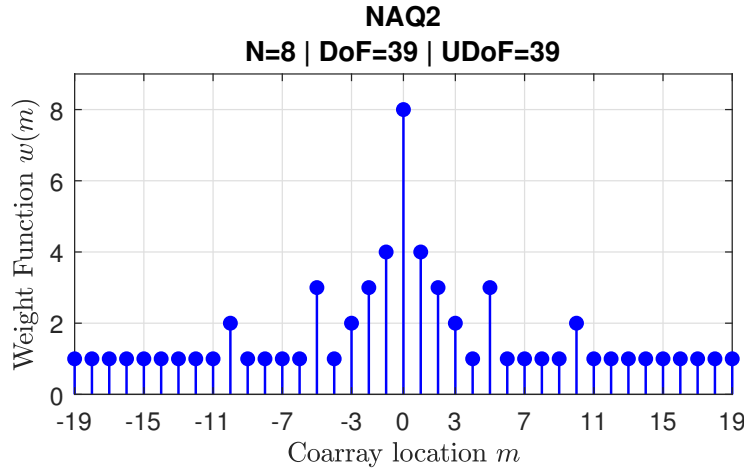


2.4(b): Weight function with UDoF = DoF = 39

Figure 2.4: NAQ2 with $N = 8$ sensors

## 2.3.2.2
## Coprime Arrays

Coprime Arrays (CPA) comprise a set $\mathbb{S}$ with formation rule dictated by

$$\mathbb{S} = \{nN_1, \ n = 0, \ldots, N_2 - 1\} \cup$$
$$\{mN_2, \ m = 1, \ldots, 2N_1 - 1\}, \quad N_2 > N_1. \quad (2\text{-}30)$$

That arrangement consists of a slightly modified version of the original CPA structure [9]. It was firstly introduced in [33]. That modified version was required to allow the use of SS-MUSIC for DoA estimation.

It is worth noticing that the cardinality of the set $\mathbb{S}$ is $|\mathbb{S}| = N_2 + 2N_1 - 1$, which clearly states that CPA has a total of $N = N_2 + 2N_1 - 1$ physical sensors. The difference coarray associated with CPA geometries have some *holes*. However, it will always possess a central uniform segment with UDoF $\geq$

$2N_1 N_2 + 1$, which allows the employment of SS-MUSIC algorithm in the same manner as for the previously presented NAQ2 arrays.

Figure 2.5 gives us an example of a CPA geometry with $N = 8$ sensors, representing the weight function with a central uniform segment of UDoF $= 23$ elements and a difference coarray set with DoF $= 27$.
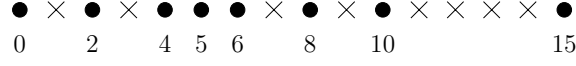


2.5(a): Geometry
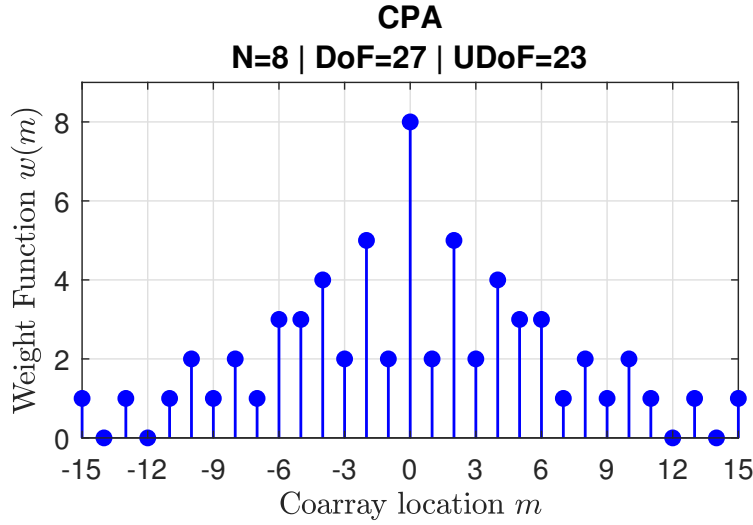


2.5(b): Weight function with UDoF $= 23$ and DoF $= 27$. Notice that the difference coarray set has holes for $m = \pm 12$ and $m = \pm 14$.

Figure 2.5: CPA with $N = 8$ sensors and $(N_1, N_2) = (2, 5)$

### 2.3.2.3
### Super Nested Arrays

The so-called 2nd-Order Super Nested Arrays (SNAQ2) were first introduced in [1] and consist of an enhanced version of the previously described NAQ2. One of the most critical features of SNAQ2 structures is the reduction in the mean of the electromagnetic coupling (EM) of the ensemble of physical sensors in terms of array design.

Therefore, when the model considers EM coupling as a disturbing side effect, usually SNAQ2 leads to better results. For the same number of sensors, these arrangements have the same aperture of that of a NAQ2 (same hole-free set $\mathbb{D}$). However, their weight function is different. This allows the employment of the SS-MUSIC algorithm to perform the multi-target DoA estimation procedure. The set $\mathbb{S}$, in this case, is given by (2-31).

$$\mathbb{S} = \mathbb{X}_1^{(2)} \cup \mathbb{Y}_1^{(2)} \cup \mathbb{X}_2^{(2)} \cup \mathbb{Y}_2^{(2)} \cup \mathbb{Z}_1^{(2)} \cup \mathbb{Z}_2^{(2)} \tag{2-31}$$

where

$$
\begin{aligned}
\mathbb{X}_1^{(2)} &= 1 + 2l \,|\, 0 \le l \le A_1 \\
\mathbb{Y}_1^{(2)} &= (N_1 + 1) - (1 + 2l) \;|\; 0 \le l \le B_1 \\
\mathbb{X}_2^{(2)} &= (N_1 + 1) + (2 + 2l) \;|\; 0 \le l \le A_2 \\
\mathbb{Y}_2^{(2)} &= 2(N_1 + 1) - (2 + 2l) \;|\; 0 \le l \le B_2 \\
\mathbb{Z}_1^{(2)} &= l(N_1 + 1) \;|\; 2 \le l \le N_2 \\
\mathbb{Z}_2^{(2)} &= N_2(N_1 + 1) - 1
\end{aligned}
\tag{2-32}
$$

The parameter set associated with (2-32) ($A_1$, $B_1$, $A_2$ and $B_2$) is defined according to (2-33).

$$
(A_1, B_1, A_2, B_2) =
\begin{cases}
(r, r-1, r-1, r-2) & \text{if } N_1 = 4r \\
(r, r-1, r-1, r-1) & \text{if } N_1 = 4r+1 \\
(r+1, r-1, r, r-2) & \text{if } N_1 = 4r+2 \\
(r, r, r, r-1) & \text{if } N_1 = 4r+3
\end{cases}
\tag{2-33}
$$

In order to illustrate some of the features, Figure 2.6(a) shows us a generic SNAQ2 structure with $N = 8$ sensors. The associated weight function is presented in Figure 2.6(b). It brings to light the hole-free coarray and the smaller magnitude of the weights next to the origin if compared to NAQ2 (Figure 2.4(b)). Moreover, confronting Figure 2.4 and Figure 2.6, we verify the equality of aperture (same physical length) and difference coarray set extension.

### 2.3.2.4
### Minimum Redundancy Arrays

Minimum Redundancy Arrays (MRA) are designed to minimize the number of pairs of physical sensors with the same position difference, under the restriction that the associated difference coarray set is hole-free. Since many DoA estimation algorithms are based on the signal covariance matrix $\mathbf{C}_{\mathbf{x}_\mathbb{S}}$, MRA structures can be useful to them due to its capability of estimating second-order statistics with a minimum number of physical sensors.

However, one of the drawbacks of MRA is the lack of a simple formation rule for the set $\mathbb{S}$. This set is defined through exhaustive search routines and consists of an intricate combinatorial optimization problem that increases its complexity with the number of physical sensors.
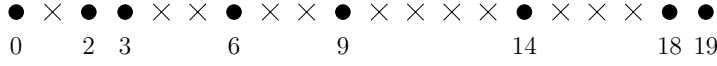
2.6(a): Geometry
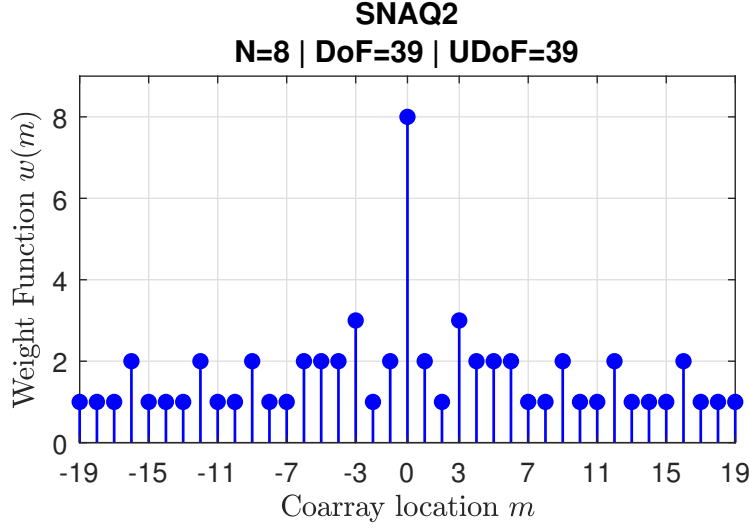


2.6(b): Weight function with UDoF = DoF = 39

Figure 2.6: SNAQ2 with $N = 8$ sensors

It can be shown that [2]:

$$N_a = \frac{N(N-1) - N_R + N_H}{2} \tag{2-34}$$

where $N_a$, $N_R$ and $N_H$ correspond to the aperture length, number of redundancies (undesired feature) and number of holes[7]. It is required that $N_H = 0$ and $N_a$ is maximized in MRA design. Some of the known MRAs presented in [2] are exhibited in Table 2.1. An MRA geometry with $N = 8$ sensors and its weight function are presented in Figure 2.7.

Table 2.1: Minimum Redundancy Arrays

| $N$ | $N_R/2$ | $N_a$ | $\mathbb{S}$ |
|---|---|---|---|
| 3 | 0 | 3 | $\{0, 1, 3\}$ |
| 4 | 0 | 6 | $\{0, 1, 4, 6\}$ |
| 5 | 1 | 9 | $\{0, 1, 4, 7, 9\}$ |
| 6 | 2 | 13 | $\{0, 1, 4, 5, 11, 13\}$ |
| 7 | 4 | 17 | $\{0, 1, 4, 10, 12, 15, 17\}$ |
| 8 | 5 | 23 | $\{0, 1, 4, 10, 16, 18, 21, 23\}$ |
| 9 | 7 | 29 | $\{0, 1, 4, 10, 16, 22, 24, 27, 29\}$ |

---

[7]The number of redundancies and holes is computed considering both sides of the weight function. The original approach in [2] considers only positive coarray locations.

2.7(a): Geometry
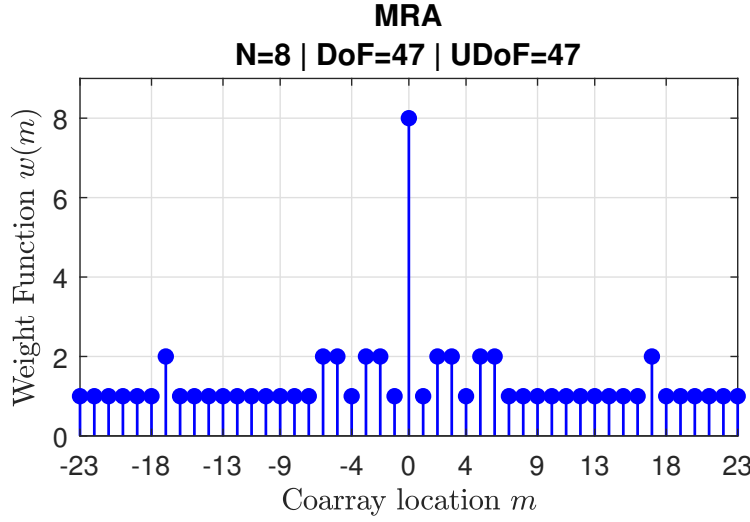


2.7(b): Weight function with UDoF = DoF = 47

Figure 2.7: MRA with $N = 8$ sensors

### 2.3.2.5
### Minimum Hole Arrays

Minimum Hole Arrays (MHA), Non-redundant Arrays or Golomb Rulers, are designed in such a way the number of holes in $\mathbb{D}$ is minimized with the restriction that the weight function has only 0 and 1 as possible elements, except for the point $(0, w(0) = N)$ [8]. That said, we minimize $N_H$ subject to $\{w(m) = 0$ or $w(m) = 1$ for $m \neq 0\}$. The constraint ensures that $|\mathbb{M}(m)| \leq 1$, $\forall\ m \neq 0$. This procedure forces a non-redundance acquisition system and generates a large aperture array compared to the other geometries discussed in this text [34].

These geometries are obtained from tabulated entries, as in the case of MRA. Some MHA described in [2] can be seen in Table 2.2. Figure 2.8(a) exhibits a generic MHA geometry with $N = 8$ sensors. The larger aperture with $34 - 0 = 34$ units is way superior to the other geometries apertures considered so far. Moreover, the weight function in Figure 2.8(b) also makes evident the increased DoF, which makes easier the estimation procedure by using Compressive Sensing algorithms. One of the drawbacks of this kind of structure is the nonexistence of closed-form solutions for the design of $\mathbb{S}$ [8, 35].

Table 2.2: Minimum Hole Arrays

| $N$ | $N_H/2$ | $N_a$ | $\mathbb{S}$ |
|---|---|---|---|
| 3 | 0 | 3 | $\{0, 1, 3\}$ |
| 4 | 0 | 6 | $\{0, 1, 4, 6\}$ |
| 5 | 1 | 11 | $\{0, 1, 4, 9, 11\}$ |
| 6 | 2 | 17 | $\{0, 1, 4, 10, 12, 17\}$ |
| 7 | 4 | 25 | $\{0, 1, 4, 10, 18, 23, 25\}$ |
| 8 | 6 | 34 | $\{0, 1, 4, 9, 15, 22, 32, 34\}$ |
| 9 | 8 | 44 | $\{0, 1, 5, 12, 25, 27, 35, 41, 44\}$ |
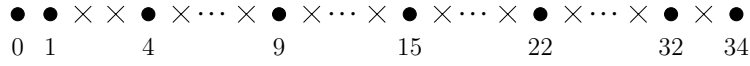


2.8(a): Geometry
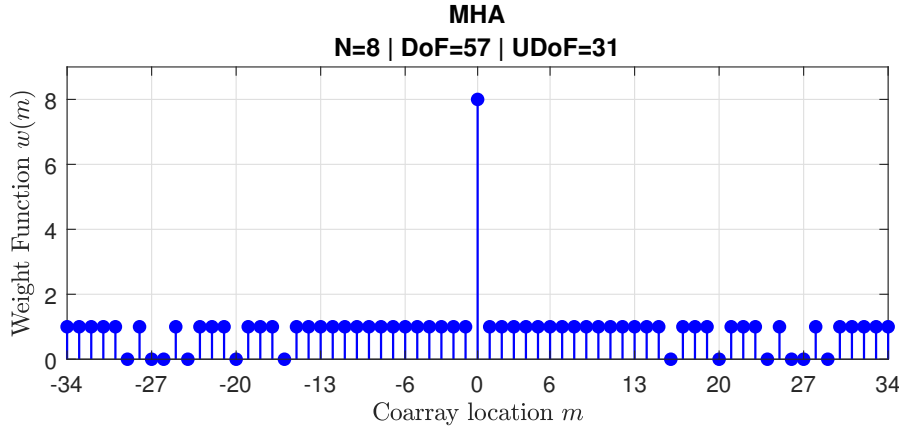


2.8(b): Weight function with UDoF $= 31$ and DoF $= 57$

Figure 2.8: MHA with $N = 8$ sensors

## 2.4
## Classic DoA Estimation Algorithms

In this section, some of the classic algorithms that are employed to perform DoA estimation are discussed, namely:

i) **Quadratic**: Beamscan (Bartlett beamformer), MVDR;

ii) **Subspace**: Spectral MUSIC; and

iii) **Stochastic Maximum Likelihood**.

### 2.4.1
### Quadratic Methods

Quadratic methods are called as such because they compute quadratic forms to estimate spatial spectra. Two of the most known algorithms of this type are the *beamscan* (Bartlett beamformer) and the *Minimum*

*Variance Distortionless Response (Capon beamformer)* or MVDR method. Each algorithm varies the steering vector $\mathbf{a}_{\mathbb{S}}(\theta)$ over the grid (range) indicated by $\boldsymbol{\theta}^g$, plots the value of a specified function and selects its $D$ peaks as the estimated DoAs $\hat{\boldsymbol{\theta}} = [\theta_1, \theta_2, \ldots, \theta_D]$. Notice that the number of sources $D$ is assumed to be known *a priori* [2].

The Bartlett beamformer computes the expression

$$\hat{P}_B(\theta) = \mathbf{a}_{\mathbb{S}}^H(\theta)\hat{\mathbf{C}}_{\mathbf{x}_{\mathbb{S}}}\mathbf{a}_{\mathbb{S}}(\theta) \tag{2-35}$$

for each $\theta$ in $\boldsymbol{\theta}_g$ and takes the $D$ largest values as corresponding to the DoA estimates [16], where $\hat{\mathbf{C}}_{\mathbf{x}_{\mathbb{S}}}$ is the sample covariance matrix defined in (2-6).

The MVDR beamformer consists of the design of a distortionless filter $\mathbf{w}$ to match a plane wave coming from $\theta$ direction [7, 15, 36, 37]. In this case, we require

$$\mathbf{w}^H\mathbf{a}_{\mathbb{S}}(\theta) = 1 \tag{2-36}$$

Furthermore, we assume the input signal covariance matrix is available. This design procedure intend to minimize the total output power subject to (2-36). This optimization leads to the MVDR filter which is given by

$$\mathbf{w}_{\mathrm{MVDR}}^H = \frac{\mathbf{a}_{\mathbb{S}}^H(\theta)\hat{\mathbf{C}}_{\mathbf{x}_{\mathbb{S}}}^{-1}}{\mathbf{a}_{\mathbb{S}}^H(\theta)\hat{\mathbf{C}}_{\mathbf{x}_{\mathbb{S}}}^{-1}\mathbf{a}_{\mathbb{S}}(\theta)} \tag{2-37}$$

This weighting is used to generate the MVDR power spectrum, which is computed as follows

$$\begin{aligned}
\hat{P}_{\mathrm{MVDR}}(\theta) &= \frac{1}{T}\sum_{t=1}^{T}\left|\mathbf{w}_{\mathrm{MVDR}}^H\dot{\mathbf{x}}(t)\right|^2 \\
&= \frac{1}{T}\sum_{t=1}^{T}\left|\frac{\mathbf{a}_{\mathbb{S}}^H(\theta)\hat{\mathbf{C}}_{\mathbf{x}_{\mathbb{S}}}^{-1}}{\mathbf{a}_{\mathbb{S}}^H(\theta)\hat{\mathbf{C}}_{\mathbf{x}_{\mathbb{S}}}^{-1}\mathbf{a}_{\mathbb{S}}(\theta)}\dot{\mathbf{x}}(t)\right|^2 \\
&= \frac{1}{\mathbf{a}_{\mathbb{S}}^H(\theta)\hat{\mathbf{C}}_{\mathbf{x}_{\mathbb{S}}}^{-1}\mathbf{a}_{\mathbb{S}}(\theta)}
\end{aligned} \tag{2-38}$$

Again, as in Bartlett beamformer, we make the $D$ largest values in (2-38) correspond to the estimated DoAs. To illustrate the discussion, consider the MVDR and Bartlett spectra in Figure 2.9. Both spectra were computed from (2-35) and (2-38). The peaks in $\theta_1 = -\pi/4$ and $\theta_2 = \pi/8$ radians correspond to two plane-wave signals impinging on an MRA with $N = 6$ sensors, $T = 100$ snapshots and SNR $= -1$ dB.

To conclude, consider the previous scenario evaluated with $R = 500$ Monte Carlo trials over an SNR range from $-10$ dB to $10$ dB, with a comparison between MRA and ULA geometries for each of the algorithms discussed above. The *root mean square error* (RMSE) is a standard metric
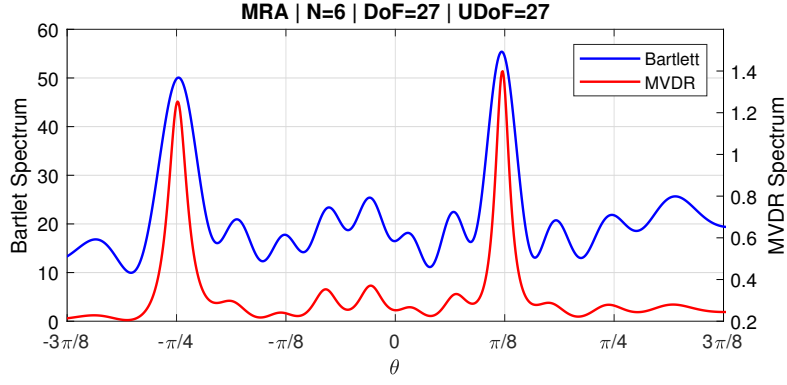
Figure 2.9: Spectrum estimate for MVDR and Bartlett methods. Two equal-power uncorrelated sources at $\theta_1 = -\pi/4$ and $\theta_2 = \pi/8$ radians, $T = 100$ snapshots, SNR $= -1$ dB. The signals impinge on an MRA with $N = 6$ sensors.

used to evaluate the algorithm performance in this case and is defined through [2]

$$\text{RMSE} = \sqrt{\frac{1}{DR}\sum_{i=1}^{R}\|\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_i\|_2^2} \tag{2-39}$$

where $R$ is the number of trials, and $\hat{\boldsymbol{\theta}}_i \in \mathbb{R}^D$ is the vector with the DoA estimation for the $D$ sources produced by the $i$-th trial.

The RMSE curves are shown in Figure 2.10. The MRA geometry leads to better performance over almost the entire SNR range. This fact is well documented in the literature: typically, NLAs have a much better performance in comparison with ULAs with the same number of array elements, once they cover a larger aperture [5, 38]. Next, subspace methods are discussed.

### 2.4.2
### Subspace-Based Methods

The subspace-based techniques for DoA estimation rely on the following properties, established from the analysis of $\mathbf{C}_{\mathbf{x}_\mathbb{S}}$ [23]:

i) The eigenstructure of that matrix comprises two subspaces spanned by its eigenvectors: the signal and noise subspaces, orthogonal to each other;

ii) The steering vectors, which are the columns of the array manifold matrix $\mathbf{A}_\mathbb{S}(\boldsymbol{\theta})$, span the signal subspace. The same space is spanned by the eigenvectors corresponding to the $D$ largest eigenvalues; and

iii) The remaining eigenvectors span the noise subspace and correspond to the $N - D$ smallest eigenvalues.

In what follows, a subspace method termed Multiple Signal Classification (MUSIC) is detailed. Let
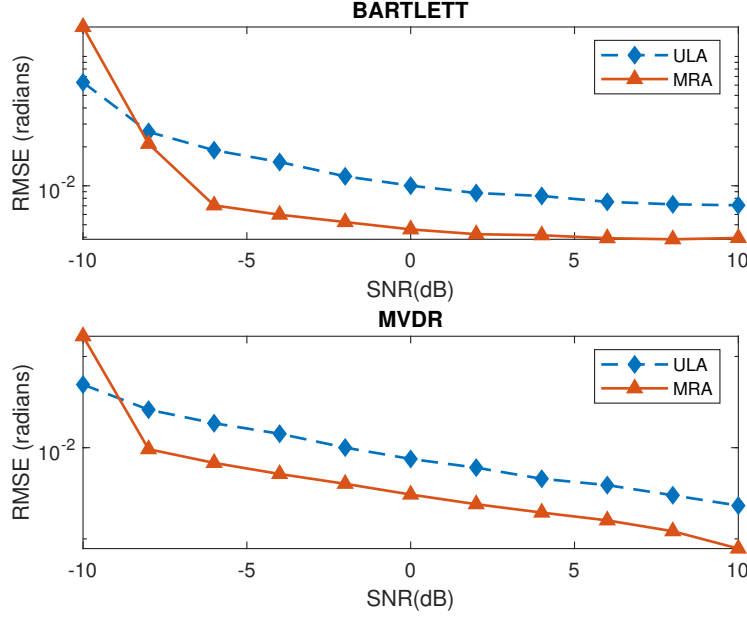
Figure 2.10: MVDR and Bartlett algorithms evaluation with $N = 6$ sensors MRA and ULA. Two equal-power uncorrelated sources at $\theta_1 = -\pi/4$ and $\theta_2 = \pi/8$, $T = 100$ snapshots and SNR $= [-10, 10]$ dB.

$$\mathbf{C}_{\mathbf{x}_{\mathbb{S}}} = \mathbf{U}\mathbf{\Sigma}\mathbf{U}^H \tag{2-40}$$

be the eigen-decomposition of the received signal covariance matrix, where

$$\begin{aligned} \mathbf{\Sigma} &= \operatorname{diag}\left(\left[\sigma_1^2 + \sigma_n^2, \ldots, \sigma_D^2 + \sigma_n^2, \sigma_n^2, \ldots, \sigma_n^2\right]\right) \\ &= \left[\begin{array}{c|c} \underset{D \times D}{\mathbf{\Sigma}_s} & \underset{D \times (N-D)}{\mathbf{0}} \\ \hline \underset{(N-D) \times D}{\mathbf{0}} & \underset{(N-D) \times (N-D)}{\mathbf{\Sigma}_n} \end{array}\right] \end{aligned} \tag{2-41}$$

is the diagonal matrix with the eigenvalues of $\mathbf{C}_{\mathbf{x}_{\mathbb{S}}}$ sorted in descending order. The matrix with the corresponding eigenvectors $\mathbf{U}$ can be written as $\mathbf{U} = [\mathbf{U}_s \mid \mathbf{U}_n]$. The range spaces of $\mathbf{U}_s \in \mathbb{C}^{N \times D}$ and $\mathbf{U}_n \in \mathbb{C}^{N \times (N-D)}$ correspond to the mutually orthogonal signal and noise subspaces, respectively. They are also linked to the largest and the smallest eigenvalues in (2-41) corresponding to the matrices $\mathbf{\Sigma}_s$ and $\mathbf{\Sigma}_n$.

Following that, $\mathcal{R}\left(\mathbf{U}_s\right)$, the range space of $\mathbf{U}_s$, equals the space spanned by the source signals [5]

$$\mathcal{R}\left(\mathbf{U}_s\right) = \mathcal{R}\left(\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})\right) \tag{2-42}$$

Since $\mathbf{U}$ is unitary, $\mathbf{U}^H\mathbf{U} = \mathbf{I}$. From that, $[\mathbf{U}_s \mid \mathbf{U}_n]^H [\mathbf{U}_s \mid \mathbf{U}_n] = \mathbf{I} \Rightarrow \mathbf{U}_s^H\mathbf{U}_n = \mathbf{0}$, which shows us that both subspaces are orthogonal. Taking that into account and considering (2-42), it follows that
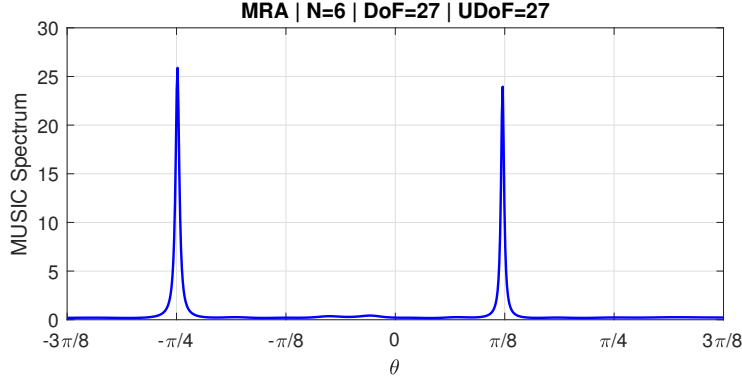
Figure 2.11: Spectrum MUSIC. Two equal-power uncorrelated sources at $\theta_1 = -\pi/4$ and $\theta_2 = \pi/8$ radians, $T = 100$ snapshots, SNR $= -1$ dB. The signals impinge on an MRA with $N = 6$ sensors.

$$\mathbf{a}_{\mathbb{S}}^{H}\left(\theta_i\right)\mathbf{U}_n = 0, \quad \text{for } i = 1, \ldots, D \tag{2-43}$$

After that, we compute

$$\hat{P}_{MU}\left(\theta\right) = \frac{1}{\mathbf{a}_{\mathbb{S}}^{H}\left(\theta_i\right)\mathbf{U}_n\mathbf{U}_n^{H}\mathbf{a}_{\mathbb{S}}\left(\theta_i\right)} \tag{2-44}$$

for all $\theta_i \in \boldsymbol{\theta}^g$, plot (2-44) (MUSIC spectrum) against $\boldsymbol{\theta}^g$ and find the largest $D$ local maxima that correspond to the $D$ estimated DoAs. Notice that Spectrum MUSIC algorithm is applicable to both NLA and ULA as well [2].

To demonstrate, consider what is shown in Figure 2.11, under the same scenario of Subsection 2.4.1. The MUSIC spectrum under the same conditions regarding number of snapshots, SNR and MRA geometry was computed. Notice that the largest peaks are located at $\theta_1 = -\pi/4$ and $\theta_2 = \pi/8$ radians, indicating the two sources impinging on that array in the corresponding directions.

Figure 2.12 shows MUSIC performance with RMSE curves against SNR for MRA and ULA with $N = 6$ sensors, highlighting the superior performance of non-uniform arrays for subspace methods, except for lower SNRs.

### 2.4.3
### Maximum Likelihood Estimators

The Maximum Likelihood Estimator (MLE) is an unbiased estimator under certain conditions. Asymptotically[8], its performance approaches the Cramér-Rao Lower Bound (CRLB). Moreover, according to what is well established in the literature [2, 39], it also possesses a satisfactory performance outside the asymptotic region.

[8]$T$, the number of snapshots, tends to $\infty$.
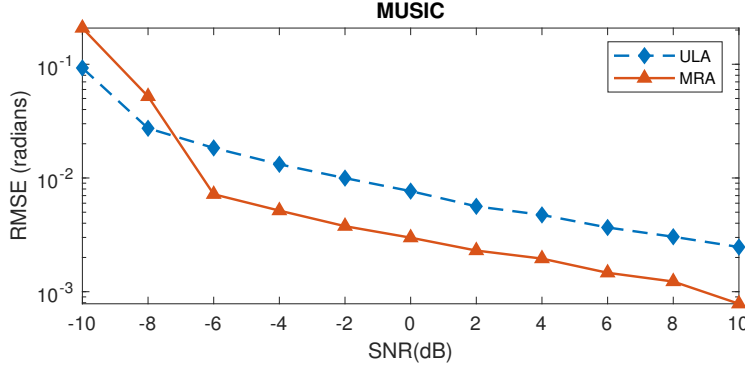
Figure 2.12: MUSIC algorithm evaluation under MRA and ULA with $N = 6$ sensors each. Two equal-power uncorrelated sources at $\theta_1 = -\pi/4$ and $\theta_2 = \pi/8$, $T = 100$ snapshots and SNR $= [-10, 10]$ dB.

Since the arriving signals in the array, under the model assumptions considered in this text, are considered to be unknown complex Gaussian distributed, the unconditional (stochastic) version of the ML estimator is considered.

Consider an arbitrary array geometry with $N$ sensors. In this case, the single snapshot probabilistic model assumed by the received signal[9] $\mathbf{x} \sim \mathcal{CN}(\mathbf{0}|\mathbf{C_x})$ is governed by the following probability density function (PDF) [2]

$$p_{\mathbf{x}(t)|\mathbf{u}}\left(\dot{\mathbf{x}}(t)\right) = \frac{1}{|\pi\mathbf{C_x}|} \exp\left\{-\dot{\mathbf{x}}^H(t)\mathbf{C_x}^{-1}\dot{\mathbf{x}}(t)\right\} \tag{2-45}$$

where $\mathbf{x}(t) \in \mathbb{C}^N$ is the received signal random vector, $\dot{\mathbf{x}}(t) \in \mathbb{C}^N$ is its realization (snapshot taken at time $t$), and $\mathbf{u} = \left[\boldsymbol{\theta}^\top \mid \mathbf{p}^\top\right]^\top \in \mathbb{R}^{2D}$ is the nonrandom unknown parameter vector to be estimated (true source directions and powers).

Notice that the received signal covariance matrix in (2-5) depends on the DoA angles $\boldsymbol{\theta}$ and the sources covariance matrix diag $(\mathbf{p})$, in such a way that a total of $2D$ parameters must be estimated, under the hypothesis that $\sigma_n^2$ is known *a priori*.

Later on, the scenario with unknown $\sigma_n^2$ is considered. For the multiple measurements vector (MMV) model, associated with independent and identically distributed (i.i.d) snapshots, the joint PDF for the total of T snapshots is given by

$$p_{\mathbf{x}(1),\ldots,\mathbf{x}(T)|\mathbf{u}}\left(\dot{\mathbf{x}}\right) = \prod_{t=1}^{T} \frac{1}{|\pi\mathbf{C_x}|} \exp\left\{-\dot{\mathbf{x}}^H(t)\mathbf{C_x}^{-1}\dot{\mathbf{x}}(t)\right\} \tag{2-46}$$

[9]To simplify the notation, we dropped the dependence of $\mathbf{x}$ on $\mathbb{S}$.

The log-likelihood function is written as

$$
\begin{aligned}
\mathcal{L}_{\mathbf{x}}\left(\mathbf{u}\right) &= \ln\left(p_{\mathbf{x}(1),\ldots,\mathbf{x}(T)|\mathbf{u}}\left(\dot{\mathbf{x}}\right)\right) \\
&= -T\ln\left(|\mathbf{C}_{\mathbf{x}}|\right) - \sum_{t=1}^{T}\dot{\mathbf{x}}^{H}(t)\mathbf{C}_{\mathbf{x}}^{-1}\dot{\mathbf{x}}(t) - TN\ln\left(\pi\right)
\end{aligned}
\tag{2-47}
$$

Pre-multiplying by $1/T$, dropping the constant term and making use of the cyclic property of the trace operator, we have

$$
\begin{aligned}
\mathcal{L}\left(\mathbf{u}\right) &= -\ln\left(|\mathbf{C}_{\mathbf{x}}|\right) - \frac{1}{T}\sum_{t=1}^{T}\dot{\mathbf{x}}^{H}(t)\mathbf{C}_{\mathbf{x}}^{-1}\dot{\mathbf{x}}(t) \\
&= -\left[\ln\left(|\mathbf{C}_{\mathbf{x}}|\right) + \mathrm{tr}\left(\frac{1}{T}\sum_{t=1}^{T}\dot{\mathbf{x}}^{H}(t)\mathbf{C}_{\mathbf{x}}^{-1}\dot{\mathbf{x}}(t)\right)\right] \\
&= -\left[\ln\left(|\mathbf{C}_{\mathbf{x}}|\right) + \mathrm{tr}\left(\mathbf{C}_{\mathbf{x}}^{-1}\frac{1}{T}\sum_{t=1}^{T}\dot{\mathbf{x}}(t)\dot{\mathbf{x}}^{H}(t)\right)\right] \\
&= -\left[\ln\left(|\mathbf{C}_{\mathbf{x}}|\right) + \mathrm{tr}\left(\mathbf{C}_{\mathbf{x}}^{-1}\hat{\mathbf{C}}_{\mathbf{x}}\right)\right]
\end{aligned}
\tag{2-48}
$$

The MLE is the value of $\mathbf{u}$ that maximizes (2-48) (the dependence of $\mathbf{u}$ is implicit in $\mathbf{C}_{\mathbf{x}}(\mathbf{u})$). In order to obtain the solution to this optimization problem, once $\mathbf{u}$ is a partitioned vector and the optimization problem formulation allows it, the first step to be taken is the maximization over $\mathbf{p}$ (or $\mathbf{C}_{\mathbf{s}}$), in such a way we obtain an explicit function of $\boldsymbol{\theta}$. After that, we maximize over the remaining $D$ variables. To perform that, we follow the method described in [2, 39].

The MLE for $\mathbf{C}_{\mathbf{s}}$, which is denoted as $\hat{\mathbf{C}}_{\mathbf{s},\mathrm{ML}}$, is given by

$$
\hat{\mathbf{C}}_{\mathbf{s},\mathrm{ML}} = \mathbf{A}_{\mathbb{S}}^{\dagger}(\boldsymbol{\theta})\left(\hat{\mathbf{C}}_{\mathbf{x}} - \sigma_n^2\mathbf{I}\right)\left[\mathbf{A}_{\mathbb{S}}^{\dagger}(\boldsymbol{\theta})\right]^{H}
\tag{2-49}
$$

where $\mathbf{A}_{\mathbb{S}}^{\dagger}(\boldsymbol{\theta}) = \left[\mathbf{A}_{\mathbb{S}}^{H}(\boldsymbol{\theta})\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})\right]^{-1}\mathbf{A}_{\mathbb{S}}^{H}(\boldsymbol{\theta})$ is the Moore-Penrose pseudoinverse associated with the matrix $\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})$. Using (2-5), we find the ML estimate for $\mathbf{C}_{\mathbf{x}}$, which will be given by

$$
\begin{aligned}
\hat{\mathbf{C}}_{\mathbf{x},\mathrm{ML}} &= \mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})\hat{\mathbf{C}}_{\mathbf{s},\mathrm{ML}}\mathbf{A}_{\mathbb{S}}^{H}(\boldsymbol{\theta}) + \sigma_n^2\mathbf{I} \\
&= \mathbf{P}_{\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})}\left(\hat{\mathbf{C}}_{\mathbf{x}} - \sigma_n^2\mathbf{I}\right)\mathbf{P}_{\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})} + \sigma_n^2\mathbf{I}
\end{aligned}
\tag{2-50}
$$

where $\mathbf{P}_{\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})} = \mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})\left[\mathbf{A}_{\mathbb{S}}^{H}(\boldsymbol{\theta})\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})\right]^{-1}\mathbf{A}_{\mathbb{S}}^{H}(\boldsymbol{\theta})$ is the orthogonal projection matrix onto the space spanned by the columns of $\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})$. Substituting (2-50) into (2-48) and manipulating the expression to find a more useful way of

expressing the final optimization problem, results in

$$\hat{\boldsymbol{\theta}}_{\mathrm{AML}} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \left\{ -\ln\left(\det\left(\mathbf{P}_{\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})}\hat{\mathbf{C}}_{\mathbf{x}}\mathbf{P}_{\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})} + \sigma_n^2 \mathbf{P}_{\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})}^{\top}\right)\right) - \right.$$
$$\left. \operatorname{tr}\left(\mathbf{P}_{\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})}^{\top}\hat{\mathbf{C}}_{\mathbf{x}}\right)/\sigma_n^2 \right\} \quad (2\text{-}51)$$

where $\det(\cdot)$ is the determinant operator, $\ln(\cdot)$ represents the natural logarithm, AML stands for *Asymptotic* ML and $\mathbf{P}_{\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})}^{\top} = \mathbf{I} - \mathbf{P}_{\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})}$ is the orthogonal projection matrix onto the null-space of $\mathbf{A}_{\mathbb{S}}^{H}(\boldsymbol{\theta})$. Since one of the model assumptions is that the variance $\sigma_n^2$ is known, for the practical case in that this quantity is not available, it can be estimated through [2]

$$\sigma_{n,\mathrm{ML}}^2 = \frac{\operatorname{tr}\left(\mathbf{P}_{\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})}^{\top}\hat{\mathbf{C}}_{\mathbf{x}}\right)}{N - D} \quad (2\text{-}52)$$

which leads to the final asymptotic MLE

$$\hat{\boldsymbol{\theta}}_{\mathrm{AML}} = \underset{\boldsymbol{\theta}}{\operatorname{argmin}} \left\{ \det\left(\mathbf{P}_{\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})}\hat{\mathbf{C}}_{\mathbf{x}}\mathbf{P}_{\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})} + \frac{\operatorname{tr}\left(\mathbf{P}_{\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})}^{\top}\hat{\mathbf{C}}_{\mathbf{x}}\right)\mathbf{P}_{\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})}^{\top}}{N - D}\right) \right\} \quad (2\text{-}53)$$

Thus, the sources DoA $\boldsymbol{\theta}$ can be estimated by maximizing (2-53), since we have at our disposal the array manifold matrix $\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})$, the number of sensors $N$, the number of sources $D$, and the sample received signal covariance matrix estimate $\hat{\mathbf{C}}_{\mathbf{x}}$, obtained through (2-6).

In general, a search procedure is performed to find the solution of (2-53). To clarify that, consider what is in Figure (2.13). The simulation scenario with MRA is again considered in this case. Since we have two parameters to be estimated, we can visualize the objective function (2-53) against the bidimensional range for each direction.

The contour plot indicates the optimal point at $\hat{\theta}_1 \approx -\pi/4$ and $\hat{\theta}_2 \approx \pi/8$. Notice that the evaluation produces a symmetric surface, which is consistent with the mathematical nature of the problem. In that way, only $g(g-1)/2$ points must be computed for the entire search. After that, optionally, a local optimization routine can be performed to precisely estimate the DoAs [2]. Notice that instead of a one-dimensional search as was done for the previous algorithms, MLE requires D-dimensional searches, which is the major drawback of this method.

## 2.5
## Difference Coarray DoA Estimation

All of the detailed algorithms described so far do not consider the difference coarray transformation in Subsection 2.2.1. Due to that, they cannot
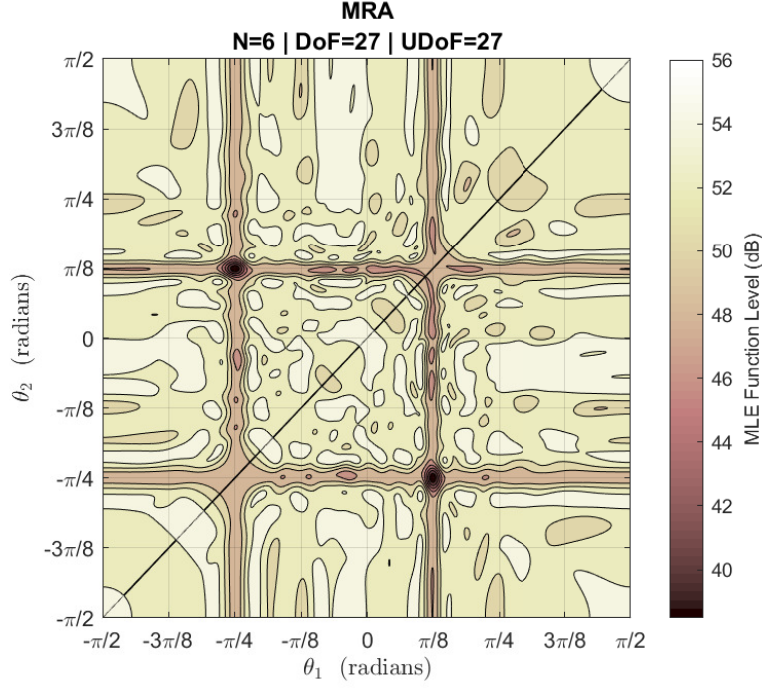
Figure 2.13: ML objective function (2-53) with the corresponding contour plot. Two equal-power uncorrelated sources at $\theta_1 = -\pi/4$ and $\theta_2 = \pi/8$, $T = 100$ snapshots, SNR $= -1$ dB. The signals impinge on an MRA with $N = 6$ sensors. Evaluation over a two-dimensional grid of $1024 \times 1024$ points in the interval $[-\pi/2, \pi/2]$. Note the optimal point in $\hat{\boldsymbol{\theta}} \approx [-\pi/4, \pi/8]^{\top}$.

take advantage of the extra DoF that arises from that kind of formulation, allowing the estimation of more sources than sensors. The following subsections describe two techniques to deal with DoA estimation in difference coarray domain: spatial smoothing MUSIC (SS-MUSIC) and compressive sensing, the latter being the focus of this work.

## 2.5.1
## Spatial Smoothing MUSIC

In this subsection, we describe the so-called *Spatial Smoothing MUSIC* (SS-MUSIC), an algorithm introduced in [17] and adapted in [40] to reduce computational complexity and turn its theoretical analysis simpler.

This algorithm performs a forward spatial smoothing based approach to build up the rank of the received signal covariance matrix in difference coarray domain. It should be remembered that spatial smoothing only can be applied to the set $\mathbb{U}$ of the difference coarray, i.e., it is required that the portion on which it is performed does not have any holes [40].

This is the main reason why SS-MUSIC can only be used with arrays that present the ratio $|\mathbb{U}| / |\mathbb{D}|$ relatively large. Ideally, it should be 1 in order
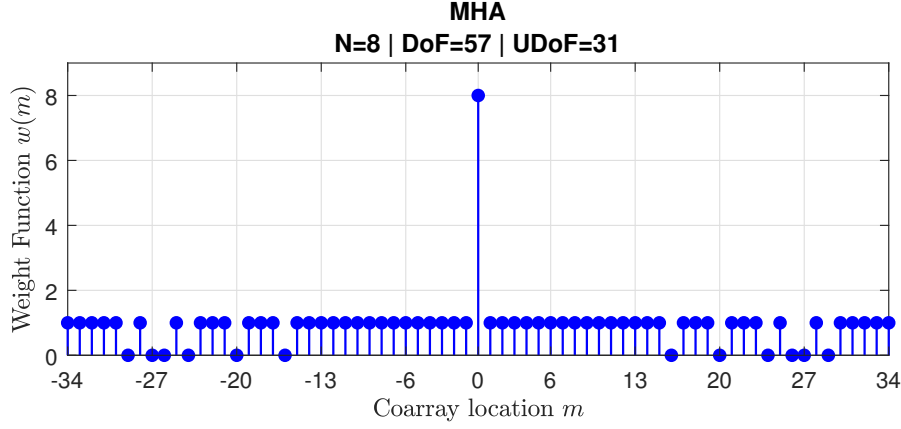
Figure 2.14: Weight function for MHA with UDoF$= 31$ and DoF$= 57$

to maximize the available DoF. Indeed, this ratio is unitary for all geometries with hole-free difference coarrays, which is the case for ULA, NAQ2, SNAQ2, and MRA.

Consider the vectorized version of $\mathbf{C_{x_S}}$ where we removed the repeated rows and sorted them in ascending order to get the sensors' positions differences in correspondence with the set $\mathbb{D}$. The resulting vector was represented as $\mathbf{x}_\mathbb{D}$ in (2-19). Define the number

$$L = \frac{|\mathbb{U}| + 1}{2} \tag{2-54}$$

as the number of sensors in each subcoarray for the spatial smoothing procedure in difference coarray domain. That said, we build the spatial smoothed matrix

$$\mathbf{R}_{ss} = \frac{1}{L} \sum_{i=0}^{L-1} \mathbf{J}_i \mathbf{x}_\mathbb{U} \left( \mathbf{J}_i \mathbf{x}_\mathbb{U} \right)^H \tag{2-55}$$

where $\mathbf{J}_i = \left[ \mathbf{0}_{L \times (L-1-i)} \mid \mathbf{I}_{L \times L} \mid \mathbf{0}_{L \times i} \right] \in \{0, 1\}^{L \times (2L-1)}$ is a selection matrix, which is positive semidefinite (PSD) [40]. This matrix is used to decompose signal and noise subspaces and allows one to estimate until $L - 1$ sources DoA. For most of the geometries, $L = \mathcal{O}\left(N^2\right)$.

To illustrate, consider again the weight function of a MHA geometry with $N = 8$ sensors in Figure 2.8(b), reproduced in Figure 2.14. It is clear that the sets $\mathbb{D}$ and $\mathbb{U}$ are given by

$$\mathbb{D} = \{0\} \cup \{\pm 1, \pm 2, \dots, \pm 15, \pm 17, \pm 18, \pm 19, \pm 21, \pm 22,$$
$$\pm 23, \pm 25, \pm 28, \pm 30, \pm 31, \dots, \pm 34\} \tag{2-56}$$

$$\mathbb{U} = \{0\} \cup \{\pm 1, \pm 2, \dots, \pm 15\} \tag{2-57}$$

These sets are graphically represented in Figure 2.15, which serves us as a guide to SS-MUSIC. Note the coarray representation in which the blue filled

square symbols represent a location for which the weight function is different from zero, i.e., the elements of $\mathbb{D}$ and $\mathbb{U} \subset \mathbb{D}$. The crosses represent *holes* (*missing lags*) in the difference coarray domain.

In the scheme, the contiguous uniform set $\mathbb{U}$ was zoomed in to highlight the useful part of the whole difference coarray for SS-MUSIC. The number of sources DoA that can be estimated by SS-MUSIC is directly connected to that contiguous central set.

Furthermore, the overlapping subcoarrays are represented as orange filled squares sliding along the set $\mathbb{U}$. Their cross-products are then used in (2-55) to build the matrix $\mathbf{R}_{ss}$ such that $\text{rank}\,(\mathbf{R}_{ss}) = L$, with an eigenstructure linked to the noise and signal subspaces, similar to what we saw in Subsection **??**.

To clarify, consider the SS-MUSIC spectrum in Figure 2.16. The simulation scenario is the same used for the classic algorithms from previous sections, except for the geometry. In this case, an MHA with $N = 8$ sensors was employed. The two peaks demonstrate the SS-MUSIC capability to detect the sources DoA. RMSE curves in Figure 2.17 demonstrate the superior performance of MHA over a ULA with the same number of sensors, following the previous comparisons.

## 2.5.2
## Compressive Sensing Algorithms

Compressive Sensing is a relatively new field of research that aims at recovering sparse solutions of underdetermined systems of linear equations with overwhelming probability (probability 1, in practice). CS algorithms are able to recover compressible (sparse) signals measured linearly, overcoming the Nyquist-Shannon sampling theorem [19, 41–43].

They have been used to undersample signals and accurately reconstruct them in many areas such as medical imaging, radar, sonar, and machine learning [19, 44]. Moreover, they have been employed to design new acquisition systems that are capable of performing a *compressed* acquisition of data. In that way, instead of obtaining tons of data and compress them after the acquisition, they allow data acquisiton in a compressed manner, so that the DAQ storage systems and A/DC converters become more straightforward and less expensive.

That said, consider the following linear measurement process

$$\mathbf{y} = \mathbf{A}\mathbf{x} \tag{2-58}$$

where $\mathbf{A} \in \mathbb{C}^{m \times N}$ is the measurement matrix with $l_2$-normalized columns, $\mathbf{y} \in \mathbb{C}^m$ is the observed data (or measurements vector) and $\mathbf{x} \in \mathbb{C}^N$ is a
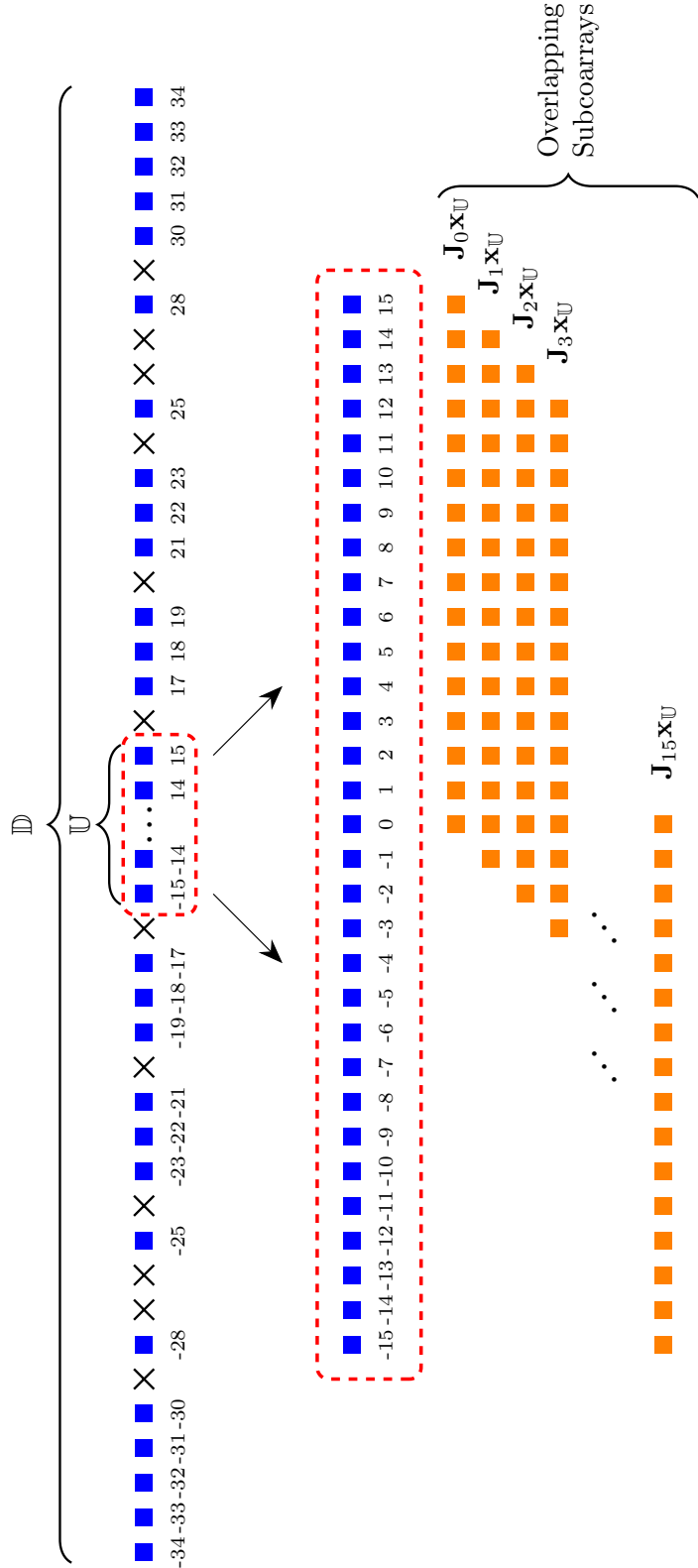
Figure 2.15: SS-MUSIC performed on MHA with $N = 8$ sensors
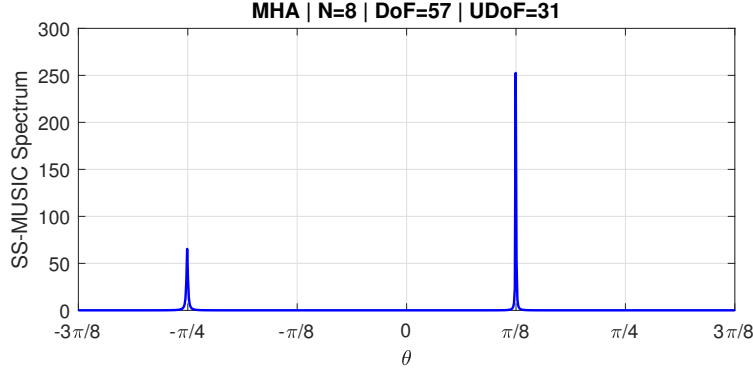
Figure 2.16: SS-MUSIC. Two equal-power uncorrelated sources at $\theta_1 = -\pi/4$ and $\theta_2 = \pi/8$, $T = 100$ snapshots, SNR $= -1$ dB. The signals impinge on an MHA with $N = 8$ sensors.
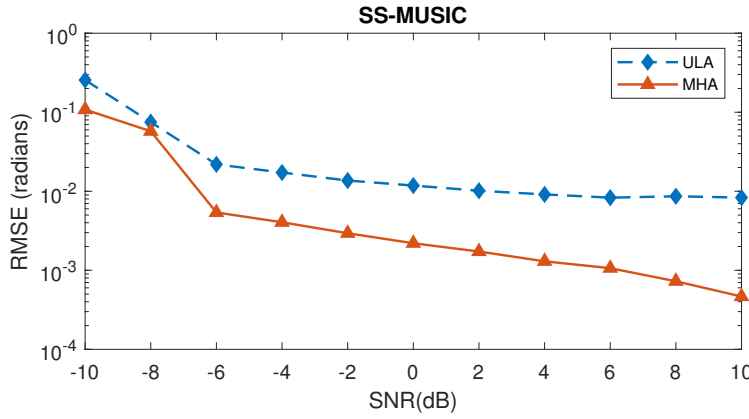


Figure 2.17: SS-MUSIC algorithm evaluation under MHA and ULA with $N = 8$ sensors each. Two equal-power uncorrelated sources at $\theta_1 = -\pi/4$ and $\theta_2 = \pi/8$, $T = 100$ snapshots and SNR $= [-10, 10]$ dB.

generic compressible (sparse) discrete signal[10], with $m < N$.

Commonly, noise measurements are considered in the model so that (2-58) becomes

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \boldsymbol{\epsilon} \qquad (2\text{-}59)$$

where typically the variable $\boldsymbol{\epsilon}$ is assumed to have the measurement noise variance in magnitude ($\|\boldsymbol{\epsilon}\|_2 = \sigma_n^2$). In this case, sparse recovery algorithms typically try to find the vector $\mathbf{x}$ that minimizes the residual Euclidean norm $\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2$.

It is worth pointing out that we can deal with signals that are not themselves sparse, but have a sparse representation under some basis $\boldsymbol{\Psi}$. In that case, $\mathbf{x} = \sum_{i=1}^{N} s_i \boldsymbol{\psi}_i = \boldsymbol{\Psi}\mathbf{s}$, where $\mathbf{s} \in \mathbb{C}^N$ is the weighting coefficients sparse vector so that $s_i = \langle \boldsymbol{\psi}_i, \mathbf{x} \rangle = \boldsymbol{\psi}_i^H \mathbf{x}$.

When this alternative basis is emphasized, the measurement matrix $\mathbf{A}$

---

[10]Notice that the variables $m$ and $N$ are generic dimensions for $\mathbf{A}$ and do not have anything to do with the number of sensors and covariance lag from previous subsections.

becomes the composition of the standard matrix product between the matrices $\mathbf{\Phi}$ and $\mathbf{\Psi}$, where $\mathbf{\Phi}$ is the fixed measurement matrix that does not depend on $\mathbf{x}$ and, for reasons to be explained later, is usually selected as a random matrix.

For example, when the signal is sparse in time domain, $\mathbf{\Psi} = \mathbf{I}$ and all the properties required to allow the sparse reconstruction imposed to $\mathbf{A}$ apply to $\mathbf{\Phi}$. To sum up, $\mathbf{A} = \mathbf{\Phi}$ or $\mathbf{A} = \mathbf{\Phi\Psi}$. In the discussion that follows, we consider the former case for the ease of representation.

Since we consider $m < N$ and $\mathbf{y} \neq \mathbf{0}$ (unknowns outnumber the equations and a DAQ system with proper functioning), the system is *nonhomogeneous underdetermined* (fewer equations than unknowns) and it is either inconsistent (no solution) or consistent with infinitely many solutions [45].

On the other hand, results in CS theory have demonstrated that if the vector $\mathbf{x}$ is *sufficiently* sparse on a given basis, which must be known *a priori*, then it can be effectively recovered from the observed data with overwhelming probability. This is called *sparse recovery* process and requires specific conditions on $\mathbf{A}$ and $\mathbf{x}$.

These conditions involve some core definitions called *Restricted Isometry Property* (RIP) and *Coherence*, which are described along with some other critical definitions in CS theory [19] [46], as follows.

**Definition 2.6** *(Overcomplete Dictionary) The linear measurement matrix $\mathbf{A}$ is defined as* overcomplete dictionary*.*

**Definition 2.7** *(Atoms) The set of columns of the matrix $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_N]$ is defined as atoms set.*

Note that, since $m < N$, the set $\{\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_N\}$ is linearly dependent, and $\text{span}(\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_N) = \mathbb{C}^m$, for a full row-rank matrix $\mathbf{A}$.

**Definition 2.8** *(Sparse Recovery Algorithm) The sparse recovery algorithm, denoted $\Delta : \mathbb{C}^m \to \mathbb{C}^N$, is a sequence of computationally tractable mathematical instructions that allows one to recover the sensed sparse signal $\mathbf{x} \in \mathbb{C}^N$ from the observed data $\mathbf{y} \in \mathbb{C}^m$.*

**Definition 2.9** *(Support) The* support *of a vector $\mathbf{x} \in \mathbb{C}^N$ is the index set of its nonzero entries, i.e.,*

$$\text{supp}(\mathbf{x}) \triangleq \{j \in [N] \mid x_j \neq 0\} \tag{2-60}$$

**Definition 2.10** *(K-Sparse Vector) The vector $\mathbf{x}$ is called K-sparse if at most K of its entries are nonzero, i.e., if*

$$\|\mathbf{x}\|_0 \triangleq |\mathrm{supp}\,(\mathbf{x})| \leq K \tag{2-61}$$

*which means that the cardinality of its support set is at most K. The notation* $\|\mathbf{x}\|_0$ *is employed to express the number of nonzero entries in the vector* $\mathbf{x}$*. Notice that this does not represent a norm at all, because it does not satisfy the homogeneity property[11].*

**Definition 2.11** *(Set of K-Sparse Vectors) The set of all K-sparse vectors is the set* $\Sigma_K$ *such that*

$$\Sigma_K \triangleq \{\mathbf{x} \mid \|\mathbf{x}\|_0 \leq K\} \tag{2-62}$$

**Definition 2.12** *(Sparsity-Level) The vector* $\mathbf{x}$ *is said to posses a sparsity-level K if its support set has cardinality K.*

**Definition 2.13** *(Coherence) Let* $\mathbf{A} \in \mathbb{C}^{m \times N}$ *be a matrix with* $l_2$*-normalized columns, i.e.,* $\|\mathbf{a}_i\|_2 = 1 \quad \forall\, i \in [N]$*. The coherence* $\mu = \mu(\mathbf{A})$ *of the matrix* $\mathbf{A}$ *is defined as*

$$\mu \triangleq \max_{1 \leq i < j \leq N} |\langle \mathbf{a}_i, \mathbf{a}_j \rangle| \tag{2-63}$$

*i.e., the largest absolute inner product between any two different columns.*

**Definition 2.14** *(Restricted Isometry Property - RIP) A matrix* $\mathbf{A}$ *satisfies the Restricted Isometry Property (RIP) of order K if there exists a* $\delta_K \in (0,1)$ *such that*

$$(1 - \delta_K)\|\mathbf{x}\|_2^2 \leq \|\mathbf{A}\mathbf{x}\|_2^2 \leq (1 + \delta_K)\|\mathbf{x}\|_2^2 \tag{2-64}$$

*holds for all* $\mathbf{x} \in \Sigma_K$

**Definition 2.15** *(Restricted Isometry Constant - RIC) The K-th restricted isometry constant* $\delta_K = \delta_K(\mathbf{A})$ *of a matrix* $A \in \mathbb{C}^{m \times N}$ *is the smallest* $\delta \geq 0$ *such that*

$$(1 - \delta_K)\|\mathbf{x}\|_2^2 \leq \|\mathbf{A}\mathbf{x}\|_2^2 \leq (1 + \delta_K)\|\mathbf{x}\|_2^2 \tag{2-65}$$

*holds for all* $\mathbf{x} \in \Sigma_K$*.*

Note that Definition 2.14 is linked to the fact that if $\mathbf{A}$ satisfies the RIP of order $2K$, then (2-64) means that the dictionary approximately preserves the distance between any chosen pair of $K$-sparse vectors. To clearly see that, it is enough to replace $\mathbf{x}$ by $\mathbf{x}_1 - \mathbf{x}_2$, with $\mathbf{x}_1, \mathbf{x}_2 \in \Sigma_K$ and, as a consequence, $\mathbf{x} \in \Sigma_{2K}$.

From that, the dimensionality reduction from $N$ to $m$ imposed on the measurement process does not confuse the vectors in the observed data space, in such a way that ambiguities are avoided during the reconstruction procedure. In general, the case of interest is when $\delta_K$ is small for reasonably large $K$. Next, we discuss two of the most popular classes of CS algorithms, namely greedy and thresholding-based methods.

---

[11]$\|a\mathbf{x}\| = |a|\,\|\mathbf{x}\|$, for all scalars $a$ (homogeneity)

### 2.5.2.1
### Greedy Algorithms

The so-called greedy algorithms are a particular class of algorithms commonly used in CS, which try to find out the sparse signal $\mathbf{x}$ support set prior to the magnitude of its non-zero coefficients. Two of the most popular CS greedy algorithms are named Orthogonal Matching Pursuit (OMP) and Subspace Pursuit (SP).

The OMP algorithm [19, 47] consists of an iterative strategy that aims at reducing the Euclidean residual norm $\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2$ at each iteration. The algorithm is formally defined in Algorithm 2.1. In this case, the sparsity level K is assumed to be known so that the algorithm performs exactly $K$ iterations and builds up a support set $\mathbb{T} = \mathrm{supp}\,(\mathbf{x})$ with cardinality $|\mathbb{T}| = K$.

There are three fundamental stages. In line 3, correlation-maximization step is performed, which extract the atom from the dictionary most correlated with the residual. This indicates one index to be merged to the sparse vector support set $\mathbb{T}$. This mergin step is carried out in line 4. After that, an orthogonal projection is performed in line 5. At the end, both the sparse signal $\mathbf{x}$ and its support set $\mathbb{T}$ are estimated as $\mathbf{x}_{\mathrm{OMP}}^{(\sharp)}$ and $\mathbb{T}^{(\sharp)}$.

---

**Algorithm 2.1:** Orthogonal Matching Pursuit - OMP

**Input** : Dictionary $\mathbf{A}$
Observed data $\mathbf{y}$
Sparsity level $K$

1   $\mathbb{T}^{(0)} = \varnothing,\ \mathbf{x}^{(0)} = \mathbf{0}$

2   **for** $i \leftarrow 0$ **to** $K-1$ **do**

3      $j \leftarrow \underset{\hat{j} \in [N]}{\mathrm{argmax}} \left\{ \left| \left[ \mathbf{A}^H \left( \mathbf{y} - \mathbf{A}\mathbf{x}^{(i)} \right) \right]_{\hat{j}} \right| \right\}$

4      $\mathbb{T}^{(i+1)} = \mathbb{T}^i \cup \{j\}$

5      $\mathbf{x}^{(i+1)} \leftarrow \underset{\mathbf{z} \in \mathbb{C}^N}{\mathrm{argmin}} \left\{ \|\mathbf{y} - \mathbf{A}\mathbf{z}\|_2,\ \mathrm{supp}\,(\mathbf{z}) \subset \mathbb{T}^{(i+1)} \right\}$

**Output:** $K$-sparse vector $\mathbf{x}_{\mathrm{OMP}}^{(\sharp)} = \mathbf{x}^{(K)}$ and support set estimate $\mathbb{T}^{(\sharp)} = \mathbb{T}^{(K)}$

---

The orthogonal projection step is a fundamental stage in many of the algorithms in CS theory. Due to that, it is worth to explain some of its fundamental aspects. Consider the scheme in Figure 2.18. We deal with the case most frequently found in practice: the noisy environment in (2-59). The observed data vector $\mathbf{y}$ is not enclosed in the dictionary range $\mathcal{R}\,(\mathbf{A})$. However, it can be decomposed into two components: $\mathbf{y}_p$ and $\mathbf{r}$, namely, the estimated noiseless observed data, entirely enclosed in $\mathcal{R}\,(\mathbf{A})$, and the residual (under perfect recovery, $\mathbf{r} = \boldsymbol{\epsilon}$).
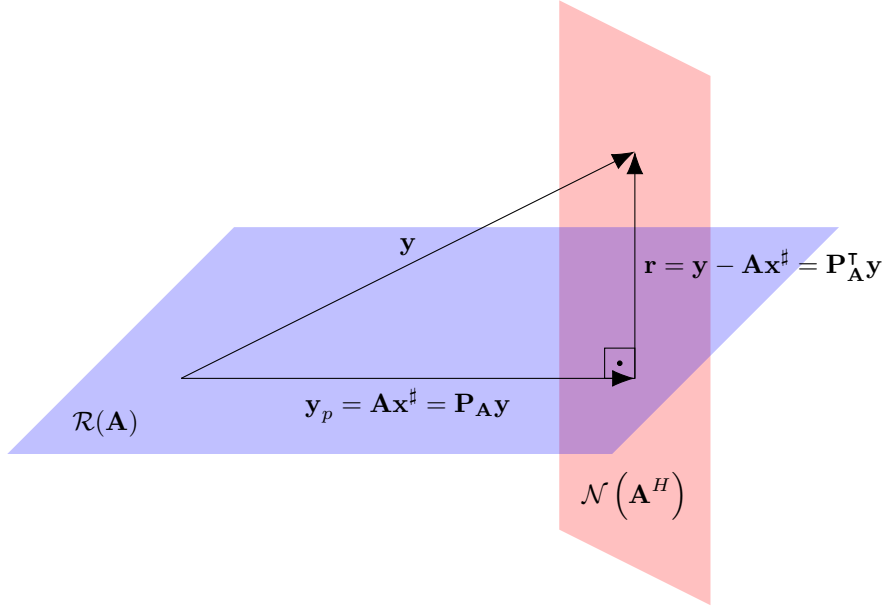
Figure 2.18: Orthogonal Projection Scheme

The former is the orthogonal projection of $\mathbf{y}$ onto $\mathcal{R}\left(\mathbf{A}\right)$. The OMP optimization in line 5 aims to estimate that vector through the search of $\mathbf{x}^{\sharp}$. Fortunately, that search has a closed-form solution given by [2]

$$
\begin{aligned}
\mathbf{x}^{\sharp} &= \left(\mathbf{A}^{H}\mathbf{A}\right)^{-1}\mathbf{A}^{H}\mathbf{y} \\
&= \mathbf{A}^{\dagger}\mathbf{y}
\end{aligned}
\tag{2-66}
$$

and

$$
\begin{aligned}
\mathbf{y}_p &= \mathbf{A}\mathbf{x}^{\sharp} \\
&= \mathbf{A}\left(\mathbf{A}^{H}\mathbf{A}\right)^{-1}\mathbf{A}^{H}\mathbf{y} \\
&= \mathbf{P_A}\mathbf{y}
\end{aligned}
\tag{2-67}
$$

where $\mathbf{A}^{\dagger} = \left(\mathbf{A}^{H}\mathbf{A}\right)^{-1}\mathbf{A}^{H}$ is the Moore-Penrose pseudoinverse of $\mathbf{A}$ and $\mathbf{P_A} = \mathbf{A}\left(\mathbf{A}^{H}\mathbf{A}\right)^{-1}\mathbf{A}^{H}$ is the orthogonal projection matrix onto the spanning of the columns of the full column rank (tall) matrix $\mathbf{A}$. The residual can be

obtained from $\mathbf{y}_p$ by

$$
\begin{aligned}
\mathbf{r} &= \mathbf{y} - \mathbf{y}_p \\
&= \mathbf{y} - \mathbf{A}\mathbf{x}^\sharp \\
&= \mathbf{y} - \mathbf{A}\left(\mathbf{A}^H\mathbf{A}\right)^{-1}\mathbf{A}^H\mathbf{y} \\
&= \left(\mathbf{I} - \mathbf{P_A}\right)\mathbf{y} \\
&= \mathbf{P_A^\top}\mathbf{y}
\end{aligned}
\tag{2-68}
$$

where $\mathbf{P_A^\top} = \mathbf{I} - \mathbf{P_A}$ is the orthogonal projection matrix onto the null space of the full row rank (fat) matrix $\mathbf{A}^H$. Moreover, let us demonstrate the orthogonality between the null and range spaces of $\mathbf{A}$.

Consider a generic vector $\mathbf{u} \in \mathcal{N}\left(\mathbf{A}^H\right)$ and $\mathbf{v} \in \mathcal{R}\left(\mathbf{A}\right)$. Then, $\mathbf{A}^H\mathbf{u} = \mathbf{0}$ and $\mathbf{v} = \mathbf{A}\mathbf{x}$. From that,

$$
\begin{aligned}
\langle \mathbf{u}, \mathbf{v} \rangle &= \mathbf{u}^H\mathbf{v} \\
&= \mathbf{u}^H\mathbf{A}\mathbf{x} \\
&= \left(\left(\mathbf{u}^H\mathbf{A}\mathbf{x}\right)^H\right)^H \\
&= \left(\mathbf{x}^H\mathbf{A}^H\mathbf{u}\right)^H \\
&= \left(\mathbf{x}^H\mathbf{0}\right) \\
&= \mathbf{0}
\end{aligned}
\tag{2-69}
$$

Consequently, if $\mathbf{r} \in \mathcal{N}\left(\mathbf{A}^H\right)$ and $\mathbf{y}_p \in \mathcal{R}\left(\mathbf{A}\right)$,

$$
\begin{aligned}
\langle \mathbf{r}, \mathbf{y}_p \rangle &= \mathbf{r}^H\mathbf{y}_p \\
&= \left(\mathbf{y}^H - \left(\mathbf{x}^\sharp\right)^H\mathbf{A}^H\right)\mathbf{A}\mathbf{x}^\sharp \\
&= \mathbf{y}^H\mathbf{A}\mathbf{x}^\sharp - \left(\mathbf{x}^\sharp\right)^H\mathbf{A}^H\mathbf{A}\mathbf{x}^\sharp \\
&= \mathbf{y}^H\mathbf{A}\mathbf{A}^\dagger\mathbf{y} - \mathbf{y}^H\left(\mathbf{A}^\dagger\right)^H\mathbf{A}^H\mathbf{A}\mathbf{A}^\dagger\mathbf{y} \\
&= \mathbf{y}^H\mathbf{A}\mathbf{A}^\dagger\mathbf{y} - \mathbf{y}^H\mathbf{A}\left(\mathbf{A}^H\mathbf{A}\right)^{-1}\left(\mathbf{A}^H\mathbf{A}\right)\mathbf{A}^\dagger\mathbf{y} \\
&= \mathbf{y}^H\mathbf{A}\mathbf{A}^\dagger\mathbf{y} - \mathbf{y}^H\mathbf{A}\mathbf{A}^\dagger\mathbf{y} \\
&= \mathbf{0}
\end{aligned}
\tag{2-70}
$$

which is consistent with (2-69). This concludes the algebraic proofs related to the geometric intuition in Figure 2.18.

Consider the following triangular brackets notation, where $\langle \mathbf{s} \rangle_{\mathbb{L}}$ is the vector in $\mathbb{C}^{|\mathbb{L}|}$ which is a restriction of $\mathbf{s}$ to the entries indexed by $\mathbb{L}$, $\langle \mathbf{A} \rangle_{\mathbb{L}}$ is the matrix in $\mathbb{C}^{m \times |\mathbb{L}|}$ which is a restriction of $\mathbf{A} \in \mathbb{C}^{m \times N}$ to the columns

indexed by $\mathbb{L}$, and $\overline{\mathbb{L}}$ is the complement of $\mathbb{L}$.

Taking that into account, the orthogonal projection solution for each iteration of Algorithm 2.1 is $\langle \mathbf{x} \rangle_{\mathbb{T}^{(i+1)}}^{(i+1)} = \langle \mathbf{A} \rangle_{\mathbb{T}^{(i+1)}}^{\dagger} \mathbf{y}$ and $\langle \mathbf{x} \rangle_{\overline{\mathbb{T}^{(i+1)}}}^{(i+1)} = \mathbf{0}$. Then, $\mathbf{x}^{(i+1)} = \langle \mathbf{x} \rangle_{\mathbb{T}^{(i+1)} \cup \overline{\mathbb{T}^{(i+1)}}}^{(i+1)}$ .

Another greedy algorithm very similar to OMP is the Subspace Pursuit (SP). This algorithm was firstly introduced in [48] and is detailed in Algorithm 2.2.

Define the operators $L_K()$ and $H_K(\cdot)$ as [19]

$$L_K(\mathbf{x}) \triangleq \text{index set of } K \text{ largest absolute values of } \mathbf{x} \qquad (2\text{-}71)$$

$$H_K(\mathbf{x}) \triangleq \mathbf{x}', \text{ where } \langle \mathbf{x}' \rangle_{L_K(\mathbf{x})} = \langle \mathbf{x} \rangle_{L_K(\mathbf{x})} \text{ and } \langle \mathbf{x}' \rangle_{\overline{L_K(\mathbf{x})}} = \mathbf{0} \qquad (2\text{-}72)$$

i.e., the operator $H_K(\cdot)$, termed K-th order hard thresholding operator, sets all but the largest (magnitude) $K$ elements of a vector to zero. For example, $L_2\left([-1, 2, -3, 5, -4]^\top\right) = \{4, 5\}$ and $H_2\left([-1, 2, -3, 5, -4]^\top\right) = [0, 0, 0, 5, -4]^\top$.

---

**Algorithm 2.2:** Subspace Pursuit - SP

**Input**  : CS Measurement matrix/dictionary $\mathbf{A}$
Observed data $\mathbf{y}$
Sparsity level $K$

1  $\mathbf{x}^{(0)} = \mathbf{0}$, $\mathbb{T}^{(0)} = \varnothing$, $i = 0$

2  **while** *some stopping criterion is not met* **do**

3     $\mathbb{V}^{(i+1)} \leftarrow L_K\left(\mathbf{A}^H\left(\mathbf{y} - \mathbf{A}\mathbf{x}^{(i)}\right)\right)$

4     $\mathbb{U}^{(i+1)} \leftarrow \mathbb{T}^{(i)} \cup \mathbb{V}^{(i+1)}$

5     $\mathbf{u}^{(i+1)} \leftarrow \underset{\mathbf{z} \in \mathbb{C}^N}{\operatorname{argmin}}\left\{\|\mathbf{y} - \mathbf{A}\mathbf{z}\|_2, \text{ supp}\,(\mathbf{z}) \subset \mathbb{U}^{(i+1)}\right\}$

6     $\mathbb{T}^{(i+1)} \leftarrow L_K\left(\mathbf{u}^{(i+1)}\right)$

7     $\mathbf{x}^{(i+1)} \leftarrow \underset{\mathbf{z} \in \mathbb{C}^N}{\operatorname{argmin}}\left\{\|\mathbf{y} - \mathbf{A}\mathbf{z}\|_2, \text{ supp}\,(\mathbf{z}) \subset \mathbb{T}^{(i+1)}\right\}$

8     $i = i + 1$

**Output:** $K$-sparse vector $\mathbf{x}_{\text{SP}}^{(\sharp)}$

---

OMP presents a lower computational complexity in comparison with SP, due to the two orthogonal projections in lines 5 and 7 of Algorithm 2.2. Notice that the last one is performed in a $K$-dimensional space for *all* the iterations. To be more specific, SP selects $K$ indices from correlation-maximization in line 3 and merges this set with the candidate support (line 4).

After that, it performs the orthogonal projection step in a range space given by the spanning of the atoms indexed by $\mathbb{U}^{(i+1)}$, according to line 5), and compute the support set estimate through the operator $L_K(\cdot)$ in line 6.

Once the support set is found, SP computes the current $K$-sparse estimation projecting the measurements onto the spanning of the atoms indexed by $\mathbb{T}^{(i+1)}$. Next, we introduce a different class of CS-based algorithms based on thresholding operations.

### 2.5.2.2
### Thresholding-Based Algorithms

The iterative hard thresholding (IHT) algorithm [49] is detailed in Algorithm 2.3. In case of successful recovery, it converges to a local minimum of the following optimization problem

$$\operatorname*{argmin}_{\mathbf{z}\in\mathbb{F}^N} \|\mathbf{y} - \mathbf{A}\mathbf{z}\|_2^2 \text{ subject to } |\text{supp}(\mathbf{z})| \leq K \qquad (2\text{-}73)$$

i.e., it seeks for the orthogonal projection solution (minimum least squares) iteratively under the constraint of a $K$-sparse signal.

The recursion in line 3 operates to solve the system $\mathbf{A}^H\mathbf{A}\mathbf{x} = \mathbf{A}^H\mathbf{y}$, which can be interpreted as the fixed-point equation $\mathbf{x}^{(i+1)} = \left(I - \mathbf{A}^H\mathbf{A}\right)\mathbf{x}^{(i)} + \mathbf{A}^H\mathbf{y} \Rightarrow \mathbf{x}^{(i+1)} = \mathbf{x}^{(i)} + \left(\mathbf{A}^H\left(\mathbf{y} - \mathbf{A}\mathbf{x}^{(i)}\right)\right)$. The $H_K(\cdot)$ operator ensures we have $K$-sparse signals, once we target $K$-sparse vectors [19, 46]. It should be noticed that IHT does not compute any orthogonal projection in each of its two steps.

---

**Algorithm 2.3:** Iterative Hard Thresholding - IHT

---
    **Input**  : CS Measurement matrix/dictionary $\mathbf{A}$, observed data $\mathbf{y}$, sparsity level $K$

1   $\mathbf{x}^{(0)} = \mathbf{0}$, $i = 0$

2   **while** *some stopping criterion is not met* **do**

3      $\mathbf{u}^{(i+1)} \leftarrow \mathbf{x}^{(i)} + \left(\mathbf{A}^H\left(\mathbf{y} - \mathbf{A}\mathbf{x}^{(i)}\right)\right)$

4      $\mathbf{x}^{(i+1)} \leftarrow H_K\left(\mathbf{u}^{(i+1)}\right)$

5      $i = i + 1$

    **Output:** $K$-sparse vector $\mathbf{x}_{\text{IHT}}^{(\sharp)}$

---

In CS theory, according to what was reviewed in Definition 2.14 and Definition 2.15, an adequate dictionary for the sparse recovery problem has to satisfy the RIP property. Gaussian random matrices are typically employed as dictionaries in CS because they satisfy RIP with high probability [41], leading to adequate rates of exact recovery.

However, due to some intrinsic characteristics of the dictionary $\mathbf{B}$ in relation to the RIP (deterministic matrix), CS algorithms need to utilize a modified version of the hard thresholding operator that sets all but the $K$ largest *local maxima* of a vector to zero.

This is slightly different from keeping the $K$ largest absolute entries because local maxima are defined as elements larger than its two neighboring entries.

As an example, consider $\mathbf{x} = [4, 5, 3, 2, 1, 2, -1]^\top$. Consequently, $H_2(\mathbf{x}) = [4, 5, 0, 0, 0, 0, 0]^\top$ and $H_2'(\mathbf{x}) = [0, 5, 0, 0, 0, 2, 0]^\top$. In case it is not possible to find $K$ local maxima, the resulting vector is padded with the largest magnitude elements in descending order. In an analogous case, $L_K'(\cdot)$ corresponds to the index set computed from the modified thresholding operation.

For example, $H_3'(\mathbf{x}) = [4, 5, 0, 0, 0, 2, 0]^\top$, $H_4'(\mathbf{x}) = [4, 5, 3, 0, 0, 2, 0]^\top$, $L_3'(\mathbf{x}) = \{1, 2, 6\}$ and $L_4'(\mathbf{x}) = \{1, 2, 3, 6\}$.

For the performed numerical experiments, it was observed that this modified operator is crucial to allow CS algorithms to converge under model (3-1) in both scenarios: the presence or absence of the EM coupling effect.

It was observed that the original versions of IHT and SP algorithms did not converge with $H_K(\cdot)/L_K(\cdot)$ for the model (2-76). Thus, for all the scenarios with curves derived from thresholding operations, the modified operators versions are employed.

## 2.5.2.3
## Denoising

Another algorithm that deserves to be described is related to the task of cleaning the noise from data. It is called *Randomized* OMP (RandOMP) and was introduced in [21]. Differently from the previous sparse recovery algorithms, this one performs multiple sparse recoveries with OMP, takes the plain average of the solutions, and employ it to obtain the cleaned signal. Consider our model in (2-59), which represents the measurement vector corrupted by noise. The model can be rewritten as

$$\mathbf{y} = \mathbf{y}' + \boldsymbol{\epsilon} \tag{2-74}$$

where the term $\mathbf{y}' = \mathbf{A}\mathbf{x}$ represents the noise-free signal. This is the quantity to be estimated by RandOMP.

At this point, it is worth emphasizing that this is *not* a sparse recovery algorithm, because the task is related to the recovery of $\mathbf{y}'$ instead of the sparse signal $\mathbf{x}$. Consider the discrete random variable (r.v) $u$ with probability mass function (PMF) given by

$$p_u(\dot{u}) = \exp\left\{ \frac{\sigma^2}{2\sigma_n^2 \left(\sigma^2 + \sigma_n^2\right)} \mathbf{a}_{\dot{u}}^H \mathbf{r}^{(i)} \right\} \tag{2-75}$$

where $\sigma$ and $\sigma_n^2$ are the sources and noise variances, supposed to be known *a priori*, $\mathbf{r}^{(i)}$ is the residue at the $i$-th iteration and $\mathbf{a}_{\dot{u}}$ is the $\dot{u}$-th atom. Instead of

choosing the most correlated atom with the residue, the choice is now random and subjected to the realization of the r.v $u$, denoted $\dot{u}$. Notice that $\dot{u} \in [N]$.

Consequently, the choice of the atoms is randomized, and the final sparse solution is stored. After we repeat this process $J_0$ times, we take the average of all the $J_0$ sparse solutions to generate a global representation that is not sparse, and project the dictionary onto it to obtain the noise-free signal. The algorithm is summarized in Algorithm 2.4.

---

**Algorithm 2.4:** Randomized Orthogonal Matching Pursuit - RandOMP

---

    **Input**  : Dictionary $\mathbf{A}$
              Observed data $\mathbf{y}$
              Sparsity level $K$
              Number of representations $J_0$

**1** $\mathbb{T}^{(0)} = \varnothing$, $\mathbf{x}^{(0)} = \mathbf{0}$, $\mathbf{r}^0 = \mathbf{y}$

**2 for** $l \leftarrow 0$ **to** $J_0 - 1$ **do**

**3**     **for** $i \leftarrow 0$ **to** $K - 1$ **do**

**4**         Take one realization of the r.v $u$ such that
        $p_u(\dot{u}) = \exp\left\{ \frac{\sigma^2}{2\sigma_n^2(\sigma^2 + \sigma_n^2)} \mathbf{a}_{\dot{u}}^H \mathbf{r}^{(i)} \right\}$

**5**         $\mathbb{T}^{(i+1)} \leftarrow \mathbb{T}^{(i)} \cup \{\dot{u}\}$

**6**         $\mathbf{x}^{(i+1)} \leftarrow \underset{\mathbf{z} \in \mathbb{C}^N}{\mathrm{argmin}} \left\{ \|\mathbf{y} - \mathbf{Az}\|_2, \ \mathrm{supp}(\mathbf{z}) \subset \mathbb{T}^{(i+1)} \right\}$

**7**         $\mathbf{r}^{(i+1)} = \mathbf{y} - \mathbf{Ax}^{(i+1)}$

**8**     $\mathbf{v}^{(l)} = \mathbf{x}^{(K)}$

**9** $y' = \mathbf{A}\left( \frac{1}{J_0} \sum_{p=0}^{J_0 - 1} v^{(p)} \right)$

    **Output:** Cleaned $\mathbf{y}'$ measurement vector

---

### 2.5.2.4
### Sparse model for difference coarray DoA estimation

Considering the CS framework presented before, we must adapt the difference coarray model in (2-19) for DoA estimation under a sparse formulation. To do that, following [18], we rewrite (2-19) as

$$
\begin{aligned}
\mathbf{x}_{\mathbb{D}} &= \mathbf{A}_{\mathbb{D}}(\boldsymbol{\theta}^g)\mathbf{p}^g + \sigma_n^2 \mathbf{i} \\
&= \left[ \mathbf{A}_{\mathbb{D}}(\boldsymbol{\theta}^g) \,\middle|\, \mathbf{i} \right] \begin{bmatrix} \mathbf{p}^g \\ \sigma_n^2 \end{bmatrix} \\
&= \mathbf{Bh}
\end{aligned}
\tag{2-76}
$$

where, $\mathbf{B} \in \mathbb{C}^{\mathrm{DoF} \times (g+1)}$, $\boldsymbol{\theta}^g \in \mathbb{R}^g$ is the grid-search vector for possible DoAs and $\mathbf{p}^g \in \mathbb{R}^g$ is the source powers augmented vector, such that $\dim(\boldsymbol{\theta}^g) =$

dim $(\mathbf{p}^g) = g$. With that, the vector $\mathbf{h} \in \mathbb{R}^{g+1}$ becomes a sparse vector with support set supp $(\mathbf{h}) = \mathbb{T}$ and the estimated DoAs are given by $\hat{\boldsymbol{\theta}} = \langle \boldsymbol{\theta}^g \rangle_{\mathbb{T} \setminus \{g+1\}}$.

## 2.6
## Summary

This chapter reviewed the data models related to DoA estimation in standard and difference coarray domain. Different non-uniform array geometries were discussed, along with the main concepts related to this field of research. Then, classic techniques for DoA estimation and its main concepts were presented, followed by a review of some algorithms in difference coarray domain like SS-MUSIC and CS algorithms. In the following chapter, we introduce a new CS algorithm specially developed to perform DoA estimation with non-uniform linear arrays.

# 3
# List-Based Maximum Likelihood Compressive Sensing Algorithm for DoA Estimation in Difference Coarray Domain

## 3.1
## Introduction

Several different approaches have been used to find the DoAs with compressive sensing algorithms [50–52]. More recently, a sparse formulation in difference coarray domain was devised to deal with NLAs [18]. In this case, the sparse solution was found through the widely known LASSO technique [53].

However, due to the sparse nature of the problem, most of the CS algorithms are suited to perform sparse recovery with this model, as is the case for the previously described OMP, IHT, SP etc [54, 55]. In this context, generic CS formulations to solve *any* sparse recovery problem are vastly covered in the literature [19].

However, since underdetermined systems of equations are commonly solved employing a fitting in terms of $l_2$-norm, when multiple candidates are considered, the rule for choosing the right candidate support is frequently associated with the $l_2$-norm of the residue [19, 51, 56]. This gap is filled with the introduction of a much more specific rule, devised in [39] for ML DoA estimation and adapted to be used with the CS framework.

This chapter takes advantage of the concepts introduced in Chapter 2 to devise a new CS algorithm for DoA estimation in the difference coarray domain. An innovative list-based algorithm with a flexible ML selection rule is presented in the context of greedy methods.

Furthermore, to complement the work, a slightly different Difference Coarray Transformation Model (DCTM) for a finite number of snapshots is proposed, followed by an application of OMP-denoising [21] in the context of non-uniform linear arrays under severe noisy conditions.

## 3.2
## Proposed List-Based ML-OMP

Consider the model for DoA estimation with CS in the difference coarray domain introduced in Subsection 2.5.2.4,

$$\mathbf{x}_{\mathbb{D}} = \left[\mathbf{A}_{\mathbb{D}}(\boldsymbol{\theta}^g) \,\middle|\, \mathbf{i}\right] \begin{bmatrix} \mathbf{p}^g \\ \sigma_n^2 \end{bmatrix} \tag{3-1}$$

$$= \mathbf{B}\mathbf{h}$$

where $\mathbf{B} \in \mathbb{C}^{\mathrm{DoF} \times (g+1)}$ is the dictionary, $\mathbf{x}_{\mathbb{D}} \in \mathbb{C}^{\mathrm{DoF}}$ is the measurement vector (received signal for the difference coarray) and $\mathbf{h} \in \mathbb{C}^{g+1}$ is the $K$-sparse vector to be recovered. In addition, if a grid $\boldsymbol{\theta}^g \in \mathbb{C}^g$ is chosen, then the vector $\mathbf{h}$ has $K = D + 1$ non-zero elements (sources and noise variances).

Clearly, from (3-1), the noise variance index is known *a priori* ($(g + 1)$-th element in $\mathbf{h}$). From that, according to our notation, $[\mathbf{h}]_{g+1} = \sigma_n^2$. Thus, if $\mathrm{supp}\,(\mathbf{h}) = \mathbb{T}$, then $g + 1 \in \mathbb{T}$ and, since $|\mathbb{T}| = D + 1$, the $D$ remaining elements of $\mathbb{T}$ must be estimated. This final task can be performed utilizing a sparse recovery algorithm that enables one to obtain the final DoA estimates through

$$\hat{\boldsymbol{\theta}} = \langle \boldsymbol{\theta}^g \rangle_{\mathbb{T} \setminus \{g+1\}} \tag{3-2}$$

Consider the OMP algorithm previously detailed in Algorithm 2.1. In order to increase the algorithm accuracy for the DoA estimation problem, a strategy based on a list of candidates for the current support set estimate was devised.

In this way, instead of choosing only one index at each iteration in the correlation-maximization (corr-max) step in line 3 of Algorithm 2.1, a list with $Q$ candidates around the corr-max index is generated. After that, the best element from the list is chosen to be the one that maximizes (2-53) over the list, i.e. MLE is used to find the estimate for the current iteration.

Due to the finite number of snapshots, modeling errors, and noise, the corr-max step in OMP sometimes fails in obtaining the proper support set element at each iteration. Thus, the rationale is that the original corr-max index must be expanded around its neighbors on the grid, increasing the chances of having the correct estimate as an element of this list with multiple candidates. Firstly, a search for the central candidate index is performed in the set

$$\mathbb{J} = [g] \setminus \mathbb{T}^{(i)} \tag{3-3}$$

arranged in ascending order, to prevent the selected atom from a previous iteration to be selected again. After that, the correlations between all the atoms in $\mathbf{B}$ are computed and the one that possesses the maximum correlation

with the residual $\mathbf{x}_{\mathbb{D}} - \mathbf{B}\mathbf{h}^{(i)}$ is selected through

$$j^{(i)} = \underset{\hat{j} \in \mathbb{J}}{\operatorname{argmax}} \left\{ \left| \left[ \mathbf{B}^H \left( \mathbf{x}_{\mathbb{D}} - \mathbf{B}\mathbf{h}^{(i)} \right) \right]_{\hat{j}} \right| \right\} \tag{3-4}$$

Consider the index $n_0$ such that

$$j^{(i)} = [\mathbb{J}]_{n_0} \tag{3-5}$$

which is a restriction on $\mathbb{J}$ to the element of this set indexed by $n_0$. From that, the set with candidate indices is compactly represented by

$$\mathbb{V} = \left\{ [\mathbb{J}]_{n_0 - n} \;\middle|\; n = \left\lfloor -\frac{Q-1}{2} \right\rfloor, \dots, \left\lceil \frac{Q-1}{2} \right\rceil \right\} \tag{3-6}$$

where $Q$ candidates are considered and $\lfloor \cdot \rfloor \, / \, \lceil \cdot \rceil$ are the *floor* and *ceil* functions, respectively. In simple terms, $\mathbb{V}$ consists of the Q nearest neighbours on $\mathbb{J}$ around $j^{(i)}$, including $j^{(i)}$ itself.

The set of candidate support sets for $\mathbf{p}^g$ is given by the union of each of the elements of $\mathbb{V}$ with the support set from the previous iteration, excluding the element $\{g + 1\}^1$, i.e.

$$\mathbb{Y}_q = \left\{ \mathbb{T}^{(i)} \setminus \{g + 1\} \right\} \cup v_q \tag{3-7}$$

where $v_q \in \mathbb{V}$ and $q \in [Q]$.

Next, consider the ML estimation in the form of the optimization problem previously described in (2-53) and recast as

$$u^{(i)} = \underset{q \in [Q]}{\operatorname{argmin}} \left\{ \det \left( \mathbf{P}_{\langle \mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}^g) \rangle_{\mathbb{Y}_q}} \hat{\mathbf{C}}_{\mathbf{x}_{\mathbb{S}}} \mathbf{P}_{\langle \mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}^g) \rangle_{\mathbb{Y}_q}} + \right. \right.$$
$$\left. \left. \frac{\operatorname{tr} \left( \mathbf{P}_{\langle \mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}^g) \rangle_{\mathbb{Y}_q}}^{\top} \hat{\mathbf{C}}_{\mathbf{x}_{\mathbb{S}}} \right)}{N - i - 1} \mathbf{P}_{\langle \mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}^g) \rangle_{\mathbb{Y}_q}}^{\top} \right) \right\} \tag{3-8}$$

where $[Q] = \{1, \dots, Q\}$ is the domain of the optimization problem (3-8) and has an one-to-one correspondence with $\{\mathbb{Y}_1, \dots, \mathbb{Y}_Q\}$. The set $\mathbb{Y}_{u^{(i)}}$ is the support set of $\mathbf{p}^g$ selected at the current iteration based on the performed ML search.

The matrices $\mathbf{P}_{\langle \mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}^g) \rangle_{\mathbb{Y}_q}}$ and $\mathbf{P}_{\langle \mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}^g) \rangle_{\mathbb{Y}_q}}^{\top} = \mathbf{I} - \mathbf{P}_{\langle \mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}^g) \rangle_{\mathbb{Y}_q}}$ are the projection matrices onto $\mathcal{R}\left( \langle \mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}^g) \rangle_{\mathbb{Y}_q} \right)$ and $\mathcal{N}\left( \langle \mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}^g) \rangle_{\mathbb{Y}_q}^H \right)$, respectively[2]. The matrix $\langle \mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}^g) \rangle_{\mathbb{Y}_q} \in \mathbb{C}^{N \times |\mathbb{Y}_q|}$ is a restriction of $\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}^g)$ to the columns

---

[1]At this point, the support set for $\mathbf{p}^g$ is referred, since the element $\{g + 1\}$ is not included. This is the support set of interest, but still has to include $\{g + 1\}$ to be the support set of $\mathbf{h}$.

[2]For a deeper coverage of some of the theoretical aspects of orthogonal projections, refer to the Subsection 2.5.2.1

indexed by $\mathbb{Y}_q$.

Then, the measurement vector $\mathbf{x}_{\mathbb{D}}$ is projected onto $\mathcal{R}(\langle \mathbf{B} \rangle_{\mathbb{T}^{(i+1)}})$ and obtain the $\left|\mathbb{T}^{(i+1)}\right|$-sparse vector $\mathbf{h}^{(i+1)}$ by means of

$$\mathbf{h}^{(i+1)} = \underset{\mathbf{z} \in \mathbb{C}^{g+1}}{\mathrm{argmin}} \left\{ \|\mathbf{x}_{\mathbb{D}} - \mathbf{Bz}\|_2, \ \mathrm{supp}\,(\mathbf{z}) \subset \mathbb{T}^{(i+1)} \right\} \qquad (3\text{-}9)$$

After $K$ iterations, the DoA estimates are then given by (3-2), where $\mathbb{T} = \mathbb{T}^{(K)}$, the final support set estimate.

LBML-OMP, summarized in Algorithm 3.1, as will be shown, has a superior performance over standard OMP for almost all the considered geometries. However, this performance improvement must take into account an increase in computational complexity.

According to [48], the computational complexity of OMP is $\mathcal{O}\left(K\mathrm{DoF}(g+1)\right)$, considering the sparse formulation of model (3-1). For that, the dominant stage is the orthogonal projection step in line 5 of Algorithm 2.1, due to the computation of the Moore-Penrose pseudoinverse.

In the case of LBML-OMP, the whole process is similar, except for the computation of the projection matrices in line 10 of Algorithm 3.1. This is the dominant stage. The evaluation through (3-8) for each candidate support set has about the same complexity as that of one OMP iteration.

This is due to the pseudoinverse calculation, according to (2-66) and (2-67). However, this is performed $Q$ times per iteration, which leads to a total computational complexity $Q + 1$ times higher than that of the standard OMP

algorithm, i.e., $\mathcal{O}\left((Q+1)K\mathrm{DoF}(g+1)\right)$.

---

**Algorithm 3.1:** List-Based Maximum Likelihood OMP (LBML-OMP)

---

**Input** : Dictionary $\mathbf{B} \in \mathbb{C}^{\mathrm{DoF}\times(g+1)}$

Observed data $\mathbf{x}_{\mathbb{D}} \in \mathbb{C}^{DoF}$

Sparsity level $K = D + 1$

Array manifold matrix over the grid $\mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}^g) \in \mathbb{C}^{N \times g}$

Sample received signal covariance matrix $\hat{\mathbf{C}}_{\mathbf{x}_{\mathbb{S}}} \in \mathbb{C}^{N \times N}$

Number of sensors $N$

Number of candidates $Q$

Grid $\boldsymbol{\theta}^g \in \mathbb{R}^g$

---

**1** $\mathbb{T}^{(0)} = \{g+1\}$, $\mathbf{h}^{(0)} = \mathbf{0}$          `// initialization`

**2** **for** $i \leftarrow 0$ **to** $K - 1$ **do**

**3**     $\mathbb{J} \leftarrow [g] \setminus \mathbb{T}^{(i)}$        `// arranged in ascending order`

**4**     $j^{(i)} \leftarrow \underset{\hat{j} \in \mathbb{J}}{\mathrm{argmax}} \left\{ \left| \left[ \mathbf{B}^H \left( \mathbf{x}_{\mathbb{D}} - \mathbf{B}\mathbf{h}^{(i)} \right) \right]_{\hat{j}} \right| \right\}$    `// max-corr step`

**5**     $n_0 \leftarrow$ position index of $j^{(i)}$ in $\mathbb{J}$

**6**     $\mathbb{V} = \left\{ [\mathbb{J}]_{n_0 - n} \ \middle| \ n = \left\lfloor -\frac{Q-1}{2} \right\rfloor, \ldots, \left\lceil \frac{Q-1}{2} \right\rceil \right\}$

**7**     **for** $q \leftarrow 1$ **to** $Q$ **do**

**8**        $v_q \leftarrow q$-th element of the set $\mathbb{V}$

**9**        $\mathbb{Y}_q = \left\{ \mathbb{T}^{(i)} \setminus \{g+1\} \right\} \cup v_q$

**10**     $u^{(i)} \leftarrow \underset{q \in [Q]}{\mathrm{argmin}} \left\{ \det \left( \mathbf{P}_{\langle \mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}^g) \rangle_{\mathbb{Y}_q}} \hat{\mathbf{C}}_{\mathbf{x}_{\mathbb{S}}} \mathbf{P}_{\langle \mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}^g) \rangle_{\mathbb{Y}_q}} + \right.\right.$

                 $\left.\left. \frac{\mathrm{tr}\left( \mathbf{P}^\top_{\langle \mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}^g) \rangle_{\mathbb{Y}_q}} \hat{\mathbf{C}}_{\mathbf{x}_{\mathbb{S}}} \right)}{N-i-1} \mathbf{P}^\top_{\langle \mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}^g) \rangle_{\mathbb{Y}_q}} \right) \right\}$

**11**     $\mathbb{T}^{(i+1)} \leftarrow \mathbb{T}^{(i)} \cup \left\{ u^{(i)} \right\}$

**12**     $\mathbf{h}^{(i+1)} \leftarrow \underset{\mathbf{z} \in \mathbb{C}^{g+1}}{\mathrm{argmin}} \left\{ \|\mathbf{x}_{\mathbb{D}} - \mathbf{B}\mathbf{z}\|_2, \ \mathrm{supp}(\mathbf{z}) \subset \mathbb{T}^{(i+1)} \right\}$

**13** $\hat{\boldsymbol{\theta}} \leftarrow \langle \boldsymbol{\theta}^g \rangle_{\mathbb{T}^{(K)} \setminus \{g+1\}}$

**Output:** Estimated Sources DoA $\hat{\boldsymbol{\theta}}$

---

## 3.3
## Enhanced Difference Coarray Transformation Model

In order to account for finite snapshot models that arise naturally in the DoA estimation problem, a slightly different formulation to the model (2-9) was devised. It was termed Enhanced Difference Coarray Transformation Model (EDCTM). Consider the following property ((A.114) in [2]) regarding the vec $(\cdot)$ operator,

$$\text{vec}\left(\mathbf{ABC}\right) = \left(\mathbf{C}^{T} \otimes \mathbf{A}\right) \text{vec}\left(\mathbf{B}\right) \tag{3-10}$$

The model for the received signal covariance matrix is given by

$$\mathbf{C}_{\mathbf{x}_\mathbb{S}} = \mathbf{A}_\mathbb{S}(\boldsymbol{\theta})\mathbf{C}_{\mathbf{s}}\mathbf{A}_\mathbb{S}^{H}(\boldsymbol{\theta}) + \mathbf{C}_{\mathbf{n}} \tag{3-11}$$

which, for spatially and temporally uncorrelated noise, becomes

$$\mathbf{C}_{\mathbf{x}_\mathbb{S}} = \mathbf{A}_\mathbb{S}(\boldsymbol{\theta})\mathbf{C}_{\mathbf{s}}\mathbf{A}_\mathbb{S}^{H}(\boldsymbol{\theta}) + \sigma_n^2\mathbf{I} \tag{3-12}$$

If the sources are uncorrelated as well, then $\mathbf{C}_{\mathbf{s}}$ is diagonal and the difference coarray model is considerably simplified by vectorizing

$$\mathbf{C}_{\mathbf{x}_\mathbb{S}} = \mathbf{A}_\mathbb{S}(\boldsymbol{\theta}) \,\text{diag}\left[\sigma_1^2, \ldots, \sigma_D^2\right] \mathbf{A}_\mathbb{S}^{H}(\boldsymbol{\theta}) + \sigma_n^2\mathbf{I} \tag{3-13}$$

That is the formulation usually found in the literature [17, 20, 57, 58]. However, evaluating (3-11) from a finite snapshot perspective, it follows that

$$\hat{\mathbf{C}}_{\mathbf{x}_\mathbb{S}} = \mathbf{A}_\mathbb{S}(\boldsymbol{\theta})\hat{\mathbf{C}}_{s}\mathbf{A}_\mathbb{S}^{H}(\boldsymbol{\theta}) + \hat{\mathbf{C}}_{n} \tag{3-14}$$

Applying the $\text{vec}\left(\cdot\right)$ operator, it leads to

$$
\begin{aligned}
\text{vec}\left(\hat{\mathbf{C}}_{\mathbf{x}_\mathbb{S}}\right) &= \text{vec}\left(\mathbf{A}_\mathbb{S}(\boldsymbol{\theta})\hat{\mathbf{C}}_{s}\mathbf{A}_\mathbb{S}^{H}(\boldsymbol{\theta})\right) + \text{vec}\left(\hat{\mathbf{C}}_{n}\right) \\
&= (\mathbf{A}_\mathbb{S}^{*}(\boldsymbol{\theta}) \otimes \mathbf{A}_\mathbb{S}(\boldsymbol{\theta})) \,\text{vec}\left(\text{diag}\left(\hat{\mathbf{C}}_{s}\right) + \hat{\mathbf{C}}_{s} - \text{diag}\left(\hat{\mathbf{C}}_{s}\right)\right) + \\
&\quad \text{vec}\left(\hat{\mathbf{C}}_{n}\right) \\
&= (\mathbf{A}_\mathbb{S}^{*}(\boldsymbol{\theta}) \otimes \mathbf{A}_\mathbb{S}(\boldsymbol{\theta})) \,\text{vec}\left(\text{diag}\left(\hat{\mathbf{C}}_{s}\right) + \underline{\hat{\mathbf{C}}}_{s}\right) + \text{vec}\left(\hat{\mathbf{C}}_{n}\right) \\
&= (\mathbf{A}_\mathbb{S}^{*}(\boldsymbol{\theta}) \otimes \mathbf{A}_\mathbb{S}(\boldsymbol{\theta})) \,\text{vec}\left(\text{diag}\left(\hat{\mathbf{C}}_{s}\right)\right) + (\mathbf{A}_\mathbb{S}^{*}(\boldsymbol{\theta}) \otimes \mathbf{A}_\mathbb{S}(\boldsymbol{\theta})) \,\text{vec}\left(\underline{\hat{\mathbf{C}}}_{s}\right) + \\
&\quad \text{vec}\left(\hat{\mathbf{C}}_{n}\right) \\
&= (\mathbf{A}_\mathbb{S}^{*}(\boldsymbol{\theta}) \circ \mathbf{A}_\mathbb{S}(\boldsymbol{\theta})) \,\text{vecd}\left(\hat{\mathbf{C}}_{s}\right) + \text{vec}\left(\mathbf{A}_\mathbb{S}(\boldsymbol{\theta})\underline{\hat{\mathbf{C}}}_{s}\mathbf{A}_\mathbb{S}^{H}(\boldsymbol{\theta})\right) + \\
&\quad \text{vec}\left(\hat{\mathbf{C}}_{n}\right)
\end{aligned}
\tag{3-15}
$$

This enables one to write

$$\text{vec}\left(\hat{\mathbf{C}}_{\mathbf{x}_\mathbb{S}}\right) - \overbrace{\text{vec}\left(\mathbf{A}_\mathbb{S}(\boldsymbol{\theta})\underline{\hat{\mathbf{C}}}_{s}\mathbf{A}_\mathbb{S}^{H}(\boldsymbol{\theta})\right)}^{\text{error term } \boldsymbol{\eta}} = (\mathbf{A}_\mathbb{S}^{*}(\boldsymbol{\theta}) \circ \mathbf{A}_\mathbb{S}(\boldsymbol{\theta})) \,\text{vecd}\left(\hat{\mathbf{C}}_{s}\right) + \quad \text{vec}\left(\hat{\mathbf{C}}_{n}\right) \tag{3-16}$$

The quantity $\boldsymbol{\eta} = \mathbf{A}_\mathbb{S}(\boldsymbol{\theta})\underline{\hat{\mathbf{C}}}_{s}\mathbf{A}_\mathbb{S}^{H}(\boldsymbol{\theta})$ will approach the null vector as the number of snapshots $T$ increases indefinitely. This is because the hollow matrix $\underline{\hat{\mathbf{C}}}_{s} \to \mathbf{0}$ when $T \to \infty$. Notice that, for practical purposes, $\hat{\mathbf{C}}_{s}$ becomes diagonal for uncorrelated sources under this asymptotic condition.

On the other hand, consider the vector $\mathbf{a}$ with its entries as the elements

of the set $\mathbb{S}$, i.e, $\mathbf{a} = [0, n_1, \ldots, n_{N-1}]^\top$. The kronecker product $\mathbf{b} = (\mathbf{a} \otimes -\mathbf{a}) \in \mathbb{R}^{N^2 \times 1}$, when indexed by a set $\mathbb{H}$, is reduced to a vector $\mathbf{b}' = \langle \mathbf{b} \rangle_{\mathbb{H}}$ such that the entries of $\mathbf{b}'$ are sorted in ascending order with removal of repeated elements. The entries of $\mathbf{b}'$ are the same as the elements of the set $\mathbb{D}$, resulting in $|\mathbb{H}| = |\mathbb{D}| = \text{DoF}$.

Consequently, row-indexing the matrix $\mathbf{A}_{\mathbb{S}}^*(\boldsymbol{\theta}) \circ \mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta})$ through $\mathbb{H}$ results in the removal of its repeated rows and sorting of the coarray locations in ascending order, i.e.,

$$\mathbf{A}_{\mathbb{D}}(\boldsymbol{\theta}) = \langle\langle \mathbf{A}_{\mathbb{S}}^*(\boldsymbol{\theta}) \circ \mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}) \rangle\rangle_{\mathbb{H}} \tag{3-17}$$

where $\langle\langle \mathbf{A}_{\mathbb{S}}^*(\boldsymbol{\theta}) \circ \mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}) \rangle\rangle_{\mathbb{H}}$ is the matrix in $\mathbb{C}^{|\mathbb{H}| \times D}$ that is a restriction of $\mathbf{A}_{\mathbb{S}}^*(\boldsymbol{\theta}) \circ \mathbf{A}_{\mathbb{S}}(\boldsymbol{\theta}) \in \mathbb{C}^{N^2 \times D}$ to the rows indexed by $\mathbb{H}$. That said, indexing both sides of (3-16) through $\mathbb{H}$, leads to

$$\hat{\mathbf{x}}_{\mathbb{D}} - \boldsymbol{\eta}' = \mathbf{A}_{\mathbb{D}}(\boldsymbol{\theta})\hat{\mathbf{p}} + \left\langle \text{vec}\left(\hat{\mathbf{C}}_n\right) \right\rangle_{\mathbb{H}} \tag{3-18}$$

Considering that $\left\langle \text{vec}\left(\hat{\mathbf{C}}_n\right) \right\rangle_{\mathbb{H}} \approx \hat{\sigma}_n^2 \mathbf{i}$, it follows that

$$\begin{aligned}
\hat{\mathbf{x}}_{\mathbb{D}} - \boldsymbol{\eta}' &= \mathbf{A}_{\mathbb{D}}(\boldsymbol{\theta})\hat{\mathbf{p}} + \hat{\sigma}_n^2 \mathbf{i} \\
&= \left[ \mathbf{A}_{\mathbb{D}}(\boldsymbol{\theta}) \,\middle|\, \mathbf{i} \right] \begin{bmatrix} \hat{\mathbf{p}} \\ \hat{\sigma}_n^2 \end{bmatrix}
\end{aligned} \tag{3-19}$$

which, after the same procedure described in Subsection 2.5.2.4, results in

$$\begin{aligned}
\hat{\mathbf{x}}_{\mathbb{D}} - \boldsymbol{\eta}' &= \mathbf{A}_{\mathbb{D}}(\boldsymbol{\theta}^g)\hat{\mathbf{p}}^g + \hat{\sigma}_n^2 \mathbf{i} \\
&= \left[ \mathbf{A}_{\mathbb{D}}(\boldsymbol{\theta}^g) \,\middle|\, \mathbf{i} \right] \begin{bmatrix} \hat{\mathbf{p}}^g \\ \hat{\sigma}_n^2 \end{bmatrix}
\end{aligned} \tag{3-20}$$

This model has a more realistic measurement vector $\hat{\mathbf{x}}_{\mathbb{D}} - \boldsymbol{\eta}'$ in comparison with DCTM. Moreover, it represents the acquired data with more precision and can be used for *any* algorithm, not only CS-based, in difference coarray DoA estimation, assuming that $\boldsymbol{\eta}'$ is known *a priori* or can be estimated from the available data. In what follows, numerical experiments demonstrate the improvements in the RMSE curves under EDCTM.

## 3.4
## Denoising as a Preprocessing Stage for DoA Estimation

In order to be able to use RandOMP denoising algorithm, which was described in Subsection 2.5.2.3, summarized in Algorithm 2.4, the model (2-1) must be adapted so that the vector with the source signals becomes sparse. To

do that, the data acquisition model must be rewritten as

$$\mathbf{x}_\mathbb{S} = \mathbf{A}_\mathbb{S}(\boldsymbol{\theta}^g)\mathbf{s}^g + \mathbf{n}_\mathbb{S} \tag{3-21}$$

where $\mathbf{A}_\mathbb{S}(\boldsymbol{\theta}^g) \in \mathbb{C}^{N \times g}$ is the extended array manifold matrix/dictionary over the grid and $\mathbf{s}^g \in \mathbb{C}^g$ is the $D$-sparse source signal vector to be recovered.

After denoising the data, a new received signal $\mathbf{x}_\mathbb{S}^{\text{RandOMP}}$ is available, cleaned from noise. This is the signal to be used in the whole process as resulting from the DAQ step. Remember that RandOMP must act on each of the available snapshots (MMV model) with the *a priori* knowledge of sources and noise powers.

## 3.5
## Results and Discussion

The following subsections exhibit the results obtained from the simulated scenarios, with the proper discussion. Different geometries and algorithms were considered. The DAQ process accounts for EM coupling and additive white Gaussian noise. The arrays are assumed to be perfectly calibrated with omnidirectional response in an isotropic and linear homogeneous medium. All the source signals correspond to uncorrelated plane waves coming from the far-field.

### 3.5.1
### Array Characterization

To pave the way towards a deeper understanding and become acquainted with the geometries and models discussed in Chapter 2, the characterization of all of the employed arrays is presented. This includes:

i) Weight functions and derived quantities (sets $\mathbb{D}$ and $\mathbb{U}$);

ii) Coupling matrices and leakage coefficients; and

iii) Uniformly weighted beampatterns.

The scenarios were built from arrays with $N = 8$ sensors, placed at positions according to specific formation rules or tabulated entries (case of MHA and MRA). The corresponding weight functions can be seen in Figure 3.1.

Notice that for all the geometries the sets $\mathbb{D}$ and $\mathbb{U}$ are the same, except for CPA and MHA. The latter has the largest aperture (largest element in $\mathbb{D}$), but with the inconvenience of having some holes. Due to that, $|\mathbb{U}|_{\text{MHA}} < |\mathbb{D}|_{\text{MHA}}$ and fewer uniform degrees of freedom are available.
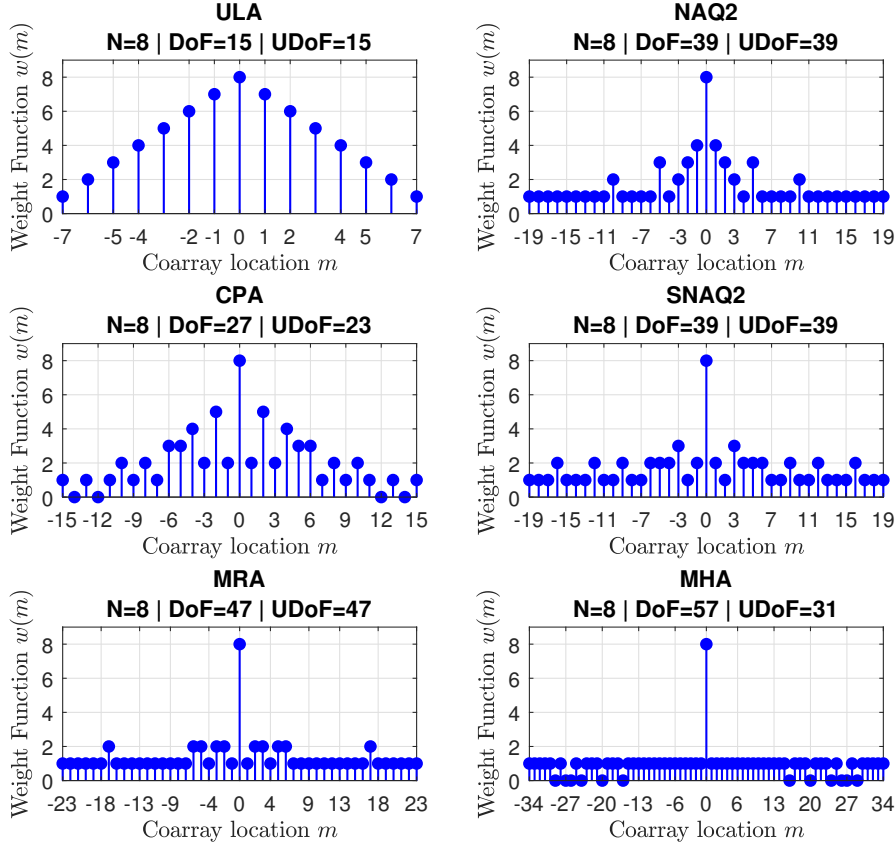
Figure 3.1: Weight function for ULA, NAQ2, CPA, SNAQ2, MRA and MHA with $N = 8$ sensors each, evidencing DoF and UDoF

Furthermore, MHA has the maximum number of degrees of freedom (DoF = 57), while MRA possesses the analogous property regarding uniform degrees of freedom (UDoF = 47). This makes sense because both the geometries result from optimization procedures that aim at finding the array that maximizes $|\mathbb{U}|$ such that $|\mathbb{D}| = |\mathbb{U}|$ (case of MRA) and minimize $(|\mathbb{D}| - |\mathbb{U}|)$ with no redundancy (unitary weights) except for the coarray location $m = 0$ (case of MHA).

On the other hand, ULA has considerably more redundancy (greater weight coefficients) with the smaller number of (uniform) degrees of freedom (DoF = UDoF = 15). Thus, in terms of redundancy and difference coarray structure, MHA/MRA and ULA are in opposite extreme sides, with MHA/MRA overcoming ULA in all possible aspects.

At this point, some drawbacks regarding MHA/MRA must be highlighted. When the constraints are critical in terms of the spatial dimension of the array, their larger aperture becomes a crucial concern. In general, in this case, one would adopt smaller aperture arrays, even ULAs, and try
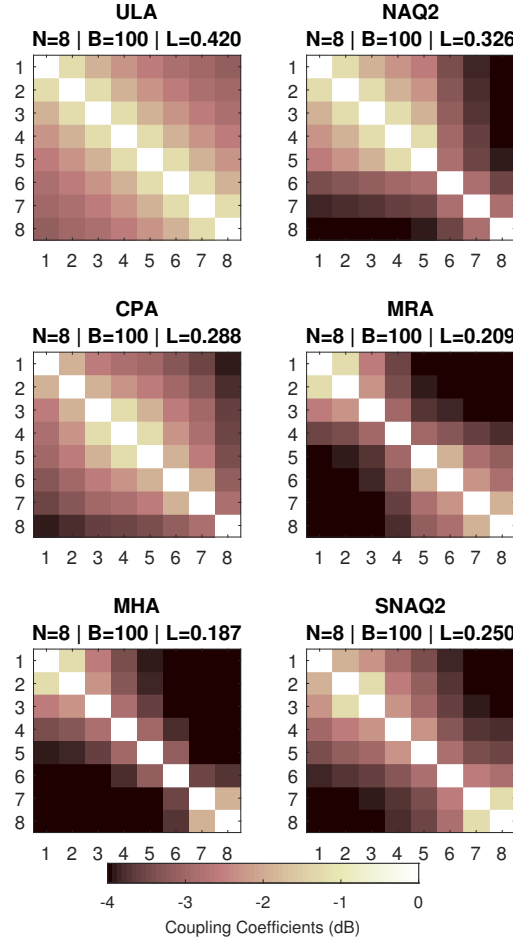
Figure 3.2: Coupling matrices and leakage coefficients for ULA, NAQ2, CPA, SNAQ2, MRA, MHA with $N = 8$ sensors each. Note that darker matrices present a smaller leakage coefficient.

to compensate for the degradation with more snapshots [2]. Moreover, for geometries obtained through computer search, as is the case for MHA/MRA, the computational complexity in array design increases dramatically when the number of physical sensors increases as well.

In terms of EM coupling, consider what is in Figure 3.2, which presents the coupling matrices $\mathbf{G}$ for each of the arrays discussed so far in a visual display, along with the leakage coefficient $L$.

In order to build the B-banded mode coupling model described in Subsection 2.2.2, the setting of parameters considers $B = 100$ and coupling coefficients generated through $c_i = c_1 \exp(-j(i-1)\pi)/i$, for $2 \leq i \leq B$ and $c_1 = 0.3 \exp(j\pi/3)$ [1]. The coefficients are pictured in dB scale. Lighter regions represent larger coupling coefficients.

In this figure, arrays that are less sensitive to EM coupling present darker coupling matrices, as is the case for MHA, MRA, and SNAQ2. Furthermore,
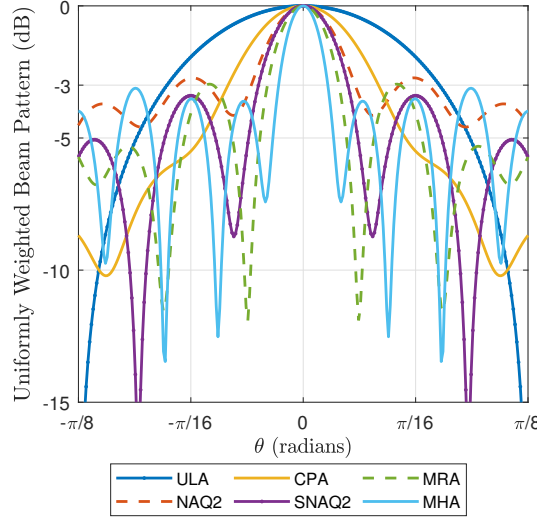
Figure 3.3: Uniformly weighted beampattern (dB scale) for ULA, NAQ2, CPA, SNAQ2, MRA, MHA with $N = 8$ sensors each, evidencing the narrower main-lobe in MHA.

these geometries have smaller leakage coefficients $L$. MHA again is the best in this sense, with $L = 0.187$. Notice that a coupling-free environment can be obtained from the setting of $B$ to zero. This would lead to diagonal coupling matrices, which have zero-valued off-diagonal coefficients.

The third aspect, which it is worth emphasizing in this array analysis, is the beampattern associated with each structure. Figure 3.3 shows the uniformly weighted beampattern zoomed in the region $[-\pi/8, \pi/8]$, so that the difference between the half-power beamwidth ($BW_{\text{HPBW}}$) (3 dB beamwidth) and beamwidth notch-notch ($BW_{\text{NN}}$) can easily be noticed for all the geometries[3]. Due to its larger aperture, MHA presents the narrower central lobe. Then, in terms of main-lobe characteristics, this suggests an improvement over all of the other geometries.

As the goal of this beampattern comparison is to establish a brief analysis, this topic will not be elaborated further. Additionally, notice that, despite uniformly weighted beampatterns being seldom used in practice, all of the arrays are compared under that same condition.

The above-mentioned characteristics of MRA/MHA justify their inclusion in the simulation scenarios, even though their design process lacks closed-form expressions [30]. Moreover, they provide a benchmark for other geometries and illustrate the performance of the algorithms acting on arrays with a larger aperture for the same number of sensors.

---

[3]Note that perfect nulls are not available. Due to that, the beamwidth notch-notch must be considered to evaluate the main lobe characteristics.

### 3.5.2
### LBML-OMP Comparison

In this subsection, RMSE curves for different geometries and multiple algorithms illustrate the superior performance of LBML-OMP. Simulation parameters include two uncorrelated plane-waves coming from $\theta_1 = -0.2952\pi$ and $\theta_2 = 0.2952\pi$ radians, $Q = 10$ candidates, $T = 400$ snapshots, and $R = 2000$ trials.

Figure 3.4 presents RMSE against SNR for the geometries considered in Subsection 3.5.1. There is a distinct enhancement in RMSE for LBML-OMP in comparison with OMP, SS-MUSIC, and IHT, except for the CPA structure.

The performance of CPA is a matter of further investigation. However, some light can be shed upon that considering the dependence of the array manifold on the set $\mathbb{S}$ and the relation between this matrix and its associated restricted isometry constants. The hypothesis laid on the assumption that the CPA structure increase the RIC for the array manifold in relation to other geometries. The initial investigation would consider this as a first conjecture.

As can be seen, none of the algorithms holds a fixed rank regarding performance when the sensing procedure employs multiple structures. However, for most of the geometries, the proposed LBML-OMP has shown superior performance.

To complement the discussion, Figure 3.5 presents the curves for RMSE against snapshots. In this case, LBML-OMP exhibits a sharp improvement over its counterparts, except for CPA, as in the previous scenario.

In addition, consider the RMSE as a multivariate function of the form

$$\text{RMSE} = f(\mathbb{S}, N, \text{SNR}, T, \Delta) \tag{3-22}$$

From that, for the same number $N$ of sensors, SNR, number of snapshots T, geometry (set $\mathbb{S}$), and two different sparse recovery algorithms ($\Delta_1$, $\Delta_2$), it is worth evaluating the region (SNR $\times$ snapshots) in which the RMSE due to OMP is higher than that resulted from LBML-OMP. Figure 3.6 shows the RMSE difference between OMP and LBML-OMP using

$$\begin{aligned}
\text{RMSE}_{\text{diff}} = \max\,(&f(\mathbb{S}, N, \text{SNR}, T, \Delta_{\text{OMP}}) \\
&- f(\mathbb{S}, N, \text{SNR}, T, \Delta_{\text{LBML-OMP}}), 0) \\
= \max\,(&\text{RMSE}_{\text{OMP}} - \text{RMSE}_{\text{LBML-OMP}}, 0)
\end{aligned} \tag{3-23}$$

where the negative values of the RMSE difference were set to zero through the $\max(\cdot, 0)$ function, in such a way it could be easier to visualize the results.

Although this kind of analysis is not usually found in the literature,
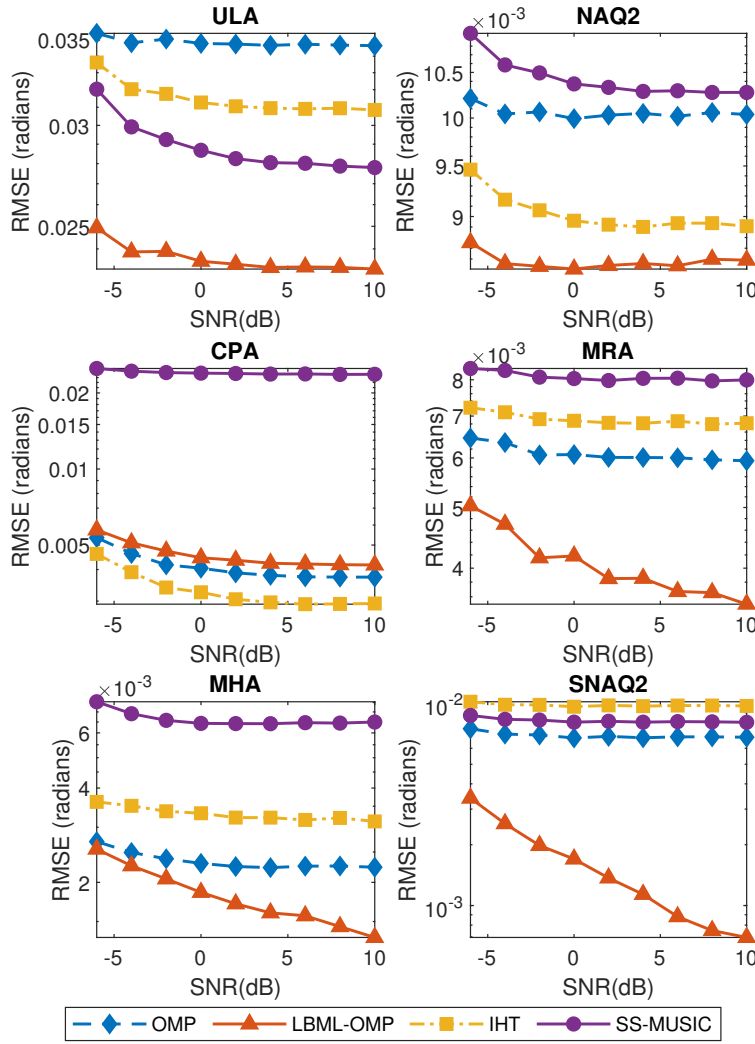
Figure 3.4: RMSE against SNR for ULA, NAQ2, CPA, SNAQ2, MRA, MHA with $N = 8$ sensors, $T = 400$ snapshots and two uncorrelated sources at $\theta_1 = -0.2952\pi$ and $\theta_2 = 0.2952\pi$ radians.

it enables one to notice that the proposed algorithm makes more significant improvements over standard OMP in regions with a greater number of snapshots and higher SNRs for all the geometries, except for CPA.

Figure 3.7 shows LBML-OMP acting on data acquired from multiple geometries. It exhibits the RMSE curves against SNR and snapshots. This is useful for investigating which geometry is the most suited for the proposed algorithm. In this case, SNAQ2.

### 3.5.3
### DCTM versus EDCTM

Figure 3.8 exhibits the RMSE curves against SNR and snapshots for an MRA geometry with $N = 8$ sensors comparing the model EDCTM developed
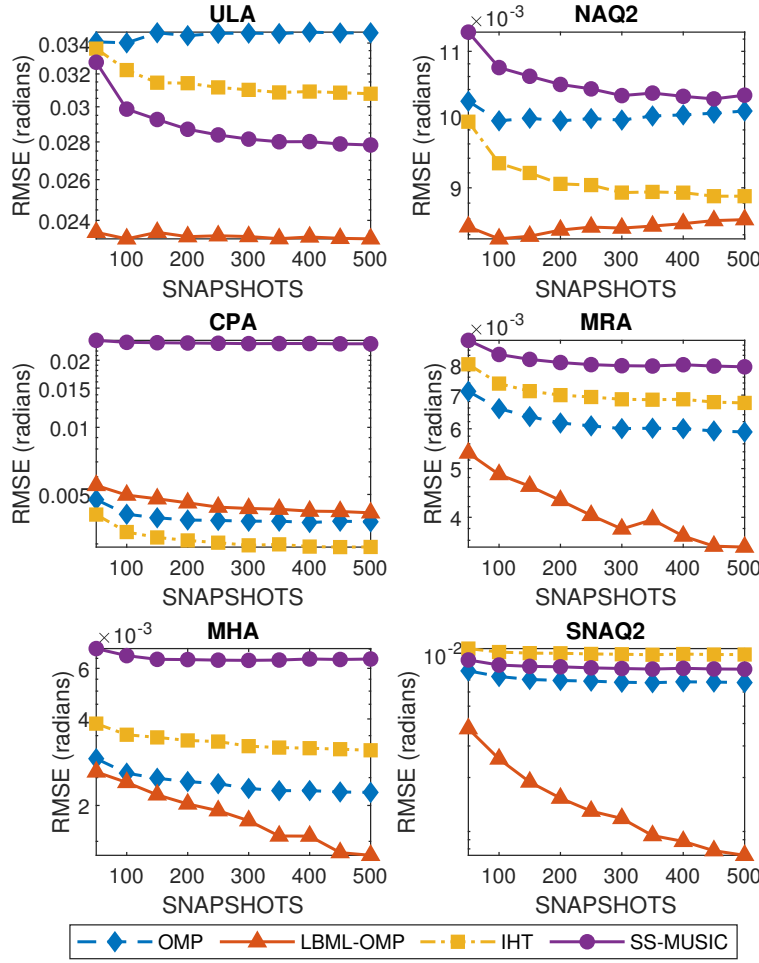
Figure 3.5: RMSE against snapshots for ULA, NAQ2, CPA, SNAQ2, MRA, MHA with $N = 8$ sensors, SNR= 6 dB and two uncorrelated sources at $\theta_1 = -0.2952\pi$ and $\theta_2 = 0.2952\pi$ radians.

in (3-19) and (3-20) (sparse version) with the traditional approach DCTM presented in (2-19) and (2-76) (sparse version) . The error term $\boldsymbol{\eta}'$ was assumed to be known *a priori*.

From these RMSE responses, it can be easily seen that the proposed model presents better results regardless of the algorithm being employed.

Moreover, notice that, in Figure 3.8, the accuracy improvement is less relevant as the number $T$ of snapshots is increased. This is because the error term $\boldsymbol{\eta}'$ in (3-19) and (3-20) tends to the null vector with the increase in $T$, once the off-diagonal elements of the sample source covariance matrix $\hat{\mathbf{C}}_{\mathbf{s}}$ tend to zero for the particular case of scenarios with uncorrelated plane waves.
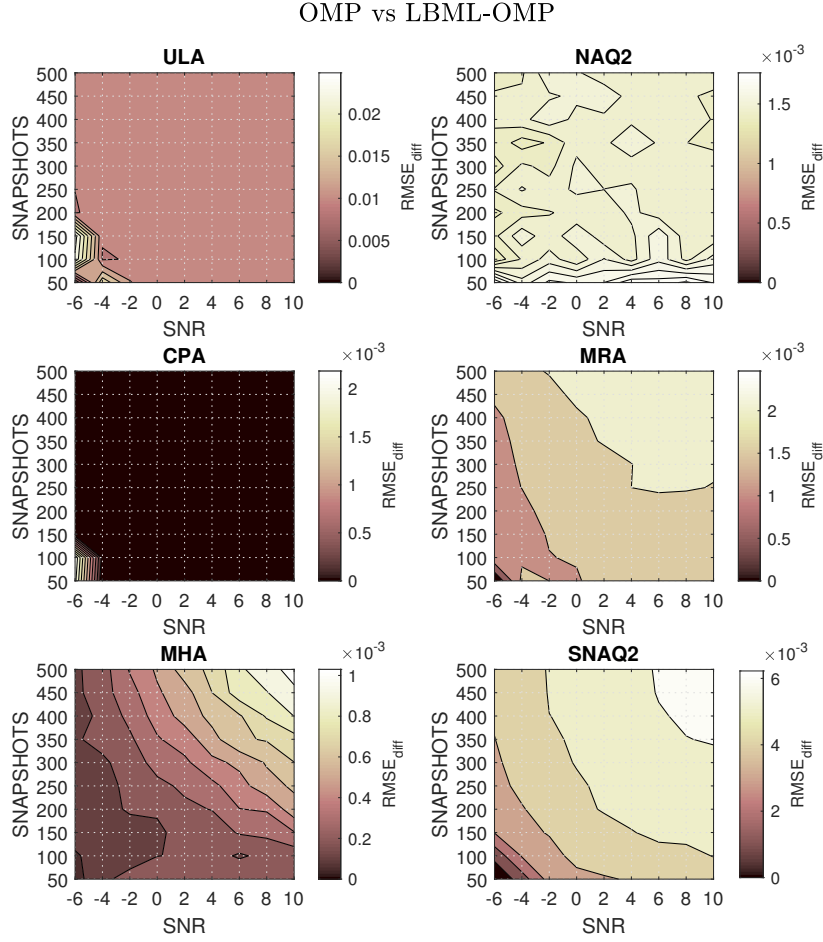
Figure 3.6: OMP vs LBML-OMP with (3-23) (contour plots for RMSE$_{\text{diff}}$, in radians). Lighter regions represent improvement of LBML-OMP over OMP. Notice that the region for which CPA give rise to suitable results is restricted to scenarios with limited number of available snapshots and low SNR.

### 3.5.4
### CS-Based Denoising in Difference Coarray Domain

Figure 3.9 shows the use of RandOMP to perform denoising on data as a preprocessing step of the DoA estimation process for SNAQ2 with $N = 8$ sensors. Two algorithms were employed: SS-MUSIC and SP. The plots consider RMSE against SNR curves at $T = 200$ snapshots. Note that the SNR range is compatible with severe noisy conditions.

From the RMSE curves, it can be ascertained that cleaning of noise from signals with RandOMP is a resourceful procedure to increase the accuracy of DoA estimation in difference coarray model. The estimation was successfully performed even for the regions that both the algorithms completely fail without this preprocessing of raw data.
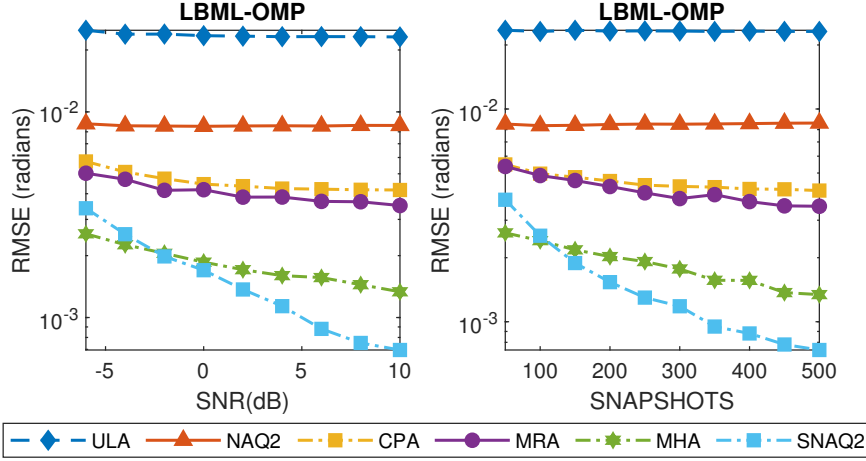
Figure 3.7: LBML-OMP acting on data acquired from multiple geometries. The RMSE responses for SNAQ2 suggest a superior performance of this algorithm with this particular array. Two uncorrelated sources at $\theta_1 = -0.2952\pi$ and $\theta_2 = 0.2952\pi$ radians. $T = 400$ snapshots (left) | SNR = 6 dB (right)
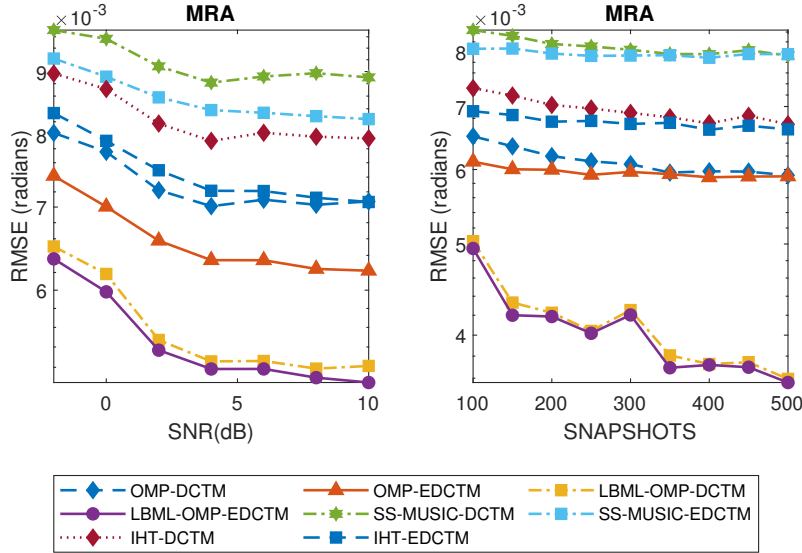


Figure 3.8: DCTM *versus* EDCTM. MRA with $N = 8$ sensors and two uncorrelated sources at $\theta_1 = -0.2952\pi$ and $\theta_2 = 0.2952\pi$ radians. $T = 50$ snapshots (left) | SNR = 4 dB (right)

## 3.6 Summary

In this chapter, a list-based maximum likelihood OMP algorithm, termed LBML-OMP, was devised. Its superior performance over standard CS algorithms and SS-MUSIC was highlighted through numerical experiments.

Furthermore, a new transformation model for difference coarray was proposed, which is more suited to fewer snapshots scenarios. In the end, RandOMP, a CS-based denoising algorithm, was used to perform cleaning
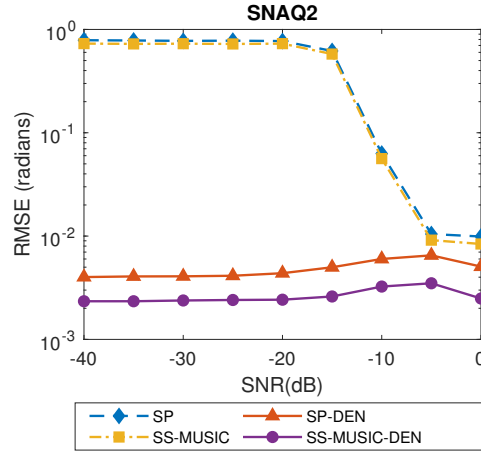
Figure 3.9: Denoising comparison. SNAQ2 with $N = 8$ sensors and two uncorrelated sources at $\theta_1 = -0.2952\pi$ and $\theta_2 = 0.2952\pi$ radians. $T = 200$ snapshots

from noise on data for severe noisy conditions as a preprocessing step in the DoA estimation procedure, presenting satisfactory results.

# 4
# Conclusions and Future Work

## 4.1
## Summary of work

In Chapter 2, data models for the DoA estimation process were presented, including the EM coupling effect. Moreover, different non-uniform arrays were detailed, along with the aspects related to their degrees of freedom and intrinsic characteristics. Then, widely known classic techniques for DoA estimation were reviewed, followed by a discussion of some algorithms that are commonly employed in difference coarray domain like SS-MUSIC and CS algorithms.

In Chapter 3, a new CS algorithm specifically designed for DoA estimation in difference coarray domain was proposed. This algorithm was shown to possesses a more accurate response in terms of RMSE for non-uniform and uniform linear arrays, except for CPA geometries. A modified version of the traditional finite snapshot model was introduced and, as a complement to the work, the task of denoising as a previous step to DoA estimation was performed with a CS-based denoising technique known as RandOMP.

## 4.2
## Future Work

The algorithm introduced in this thesis lacks a sufficient condition on the deterministic dictionary to perform sparse recovery with success in terms of RIC/RIP. Thus, it would be interesting to perform a RIP based sparse recovery condition analysis. Moreover, refined conditions at the initialization can be established to improve results.

Further analysis to be carried out in future works includes investigation in scenarios for which correlated sources are considered, with correlation coefficients of different magnitudes and phases, sources with substantially different power levels, and fractional beamwidth separations.

Another case of interest is the development of a method to decrease the coherence of the dictionary $\mathbf{B}$. This can be done in several ways, such as the introduction of random matrices in the sparse model for difference coarray

DoA estimation and optimum placement of sensors, similar to what was done in [59].

Furthermore, the match between algorithm and geometry must be investigated in order to theoretically explain some of the results empirically found in Subsection 3.5.2.

One of the premises of the development performed in this thesis is that the sources are fixed. However, the algorithms could be extended for a moving source case with different noise characteristics for each snapshot. Accounting for different measurement vector noise powers from snapshot to snapshot is a relevant matter. An interesting starting point would be the method described in [60].

A further topic of investigation is related to efficient ways of estimating the error term $\boldsymbol{\eta}'$ for the developed EDCTM model, so that the use of the proposed model cost-effectively assumes a practical form. A preliminary approach would be to try to develop an ML estimator for this term based on the ML estimator of the signal covariance matrix given in [2] and some denoising technique as RandOMP [21].

# Bibliography

[1] LIU, C.-L.; VAIDYANATHAN, P. P.. **Super Nested Arrays: Linear Sparse Arrays With Reduced Mutual Coupling—Part I: Fundamentals**. IEEE Transactions on Signal Processing, 64(15):3997–4012, aug 2016.

[2] VAN TREES, H. L.. **Optimum Array Processing**. John Wiley & Sons, Inc., New York, USA, mar 2002.

[3] QIU, L.; CAI, Y.; DE LAMARE, R. C. ; ZHAO, M.. **Reduced-Rank DOA Estimation Algorithms Based on Alternating Low-Rank Decomposition**. IEEE Signal Processing Letters, 23(5):565–569, may 2016.

[4] WANG, L.; DE LAMARE, R. C. ; HAARDT, M.. **Direction finding algorithms based on joint iterative subspace optimization**. IEEE Transactions on Aerospace and Electronic Systems, 50(4):2541–2553, oct 2014.

[5] TUNCER, T. E.; FRIEDLANDER, B.. **Classical and Modern Direction-of-Arrival Estimation**. Academic Press, Inc., Orlando, FL, USA, 2009.

[6] SHI, Z.; ZHOU, C.; GU, Y.; GOODMAN, N. A. ; QU, F.. **Source Estimation Using Coprime Array: A Sparse Reconstruction Perspective**. IEEE Sensors Journal, 17(3):755–765, feb 2017.

[7] RUAN, H.. **Efficient Robust Adaptive Beamforming Algorithms for Sensor Arrays**. PhD thesis, 2016.

[8] LIU, C.-L.; VAIDYANATHAN, P.. **Cramér–Rao bounds for coprime and other sparse arrays, which find more sources than sensors**. Digital Signal Processing, 61:43–61, feb 2017.

[9] VAIDYANATHAN, P. P.; PAL, P.. **Sparse sensing with coprime arrays**. In: 2010 CONFERENCE RECORD OF THE FORTY FOURTH ASILOMAR CONFERENCE ON SIGNALS, SYSTEMS AND COMPUTERS, p. 1405–1409. IEEE, nov 2010.

[10] PAL, P.; VAIDYANATHAN, P. P.. **Nested arrays in two dimensions, Part I: Geometrical Considerations**. IEEE Transactions on Signal Processing, 60(9):4694–4705, 2012.

[11] STOICA, P.; NEHORAI, A.. **MUSIC, maximum likelihood, and Cramer-Rao bound**. IEEE Transactions on Acoustics, Speech, and Signal Processing, 37(5):720–741, may 1989.

[12] JOHNSON, D.; DUDGEON, D.. **Array Signal Processing: Concepts and Techniques**. Prentice-Hall Signal Processing Series. P T R Prentice Hall, 1993.

[13] SCHMIDT, R.. **Multiple emitter location and signal parameter estimation**. IEEE Transactions on Antennas and Propagation, 34(3):276–280, mar 1986.

[14] ROY, R.; KAILATH, T.. **ESPRIT-estimation of signal parameters via rotational invariance techniques**. IEEE Transactions on Acoustics, Speech, and Signal Processing, 37(7):984–995, jul 1989.

[15] CAPON, J.. **High-resolution frequency-wavenumber spectrum analysis**. Proceedings of the IEEE, 57(8):1408–1418, 1969.

[16] STOICA, P.; MOSES, R.. **Spectral Analysis of Signals**. Pearson Prentice Hall, 2005.

[17] PAL, P.; VAIDYANATHAN, P. P.. **A novel array structure for directions-of-arrival estimation with increased degrees of freedom**. In: 2010 IEEE INTERNATIONAL CONFERENCE ON ACOUSTICS, SPEECH AND SIGNAL PROCESSING, número 2, p. 2606–2609. IEEE, 2010.

[18] ZHANG, Y. D.; AMIN, M. G. ; HIMED, B.. **Sparsity-based DOA estimation using co-prime arrays**. ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings, p. 3967–3971, 2013.

[19] FOUCART, S.; RAUHUT, H.. **A Mathematical Introduction to Compressive Sensing**, volumen 44 de **Applied and Numerical Harmonic Analysis**. Springer New York, New York, NY, feb 2013.

[20] SHI, J.; HU, G.; ZHOU, H. ; ZHANG, X.. **Generalized Nested Array: Optimization for Degrees of Freedom and Mutual Coupling**. IEEE Communications Letters, 22(6):1208–1211, 2018.

[21] ELAD, M.; YAVNEH, I.. **A Plurality of Sparse Representations Is Better Than the Sparsest One Alone**. IEEE Transactions on Information Theory, 55(10):4701–4714, oct 2009.

[22] NAIDU, P.. **Sensor Array Signal Processing**. Taylor & Francis, 2000.

[23] CHEN, Z.; GOKEDA, G. ; YU, Y.. **Introduction to Direction-of-arrival Estimation**. Artech House signal processing library. Artech House, 2010.

[24] BENESTY, J.; COHEN, I. ; CHEN, J.. **Fundamentals of Signal Enhancement and Array Signal Processing**. John Wiley & Sons Singapore Pte. Ltd, Singapore, dec 2017.

[25] PINTO, S. F. B.; DE LAMARE, R. C.. **Multi-step knowledge-aided iterative ESPRIT for direction finding**. In: 2017 22ND INTERNATIONAL CONFERENCE ON DIGITAL SIGNAL PROCESSING (DSP), p. 1–5. IEEE, aug 2017.

[26] SONG, N.; ALOKOZAI, W. U.; DE LAMARE, R. C. ; HAARDT, M.. **Adaptive Widely Linear Reduced-Rank Beamforming Based on Joint Iterative Optimization**. IEEE Signal Processing Letters, 21(3):265–269, mar 2014.

[27] SOMASUNDARAM, S. D.; PARSONS, N. H.; LI, P. ; DE LAMARE, R. C.. **Reduced-dimension robust capon beamforming using Krylov-subspace techniques**. IEEE Transactions on Aerospace and Electronic Systems, 51(1):270–289, jan 2015.

[28] LEV-ARI, H.. **Efficient solution of linear matrix equations with application to multistatic antenna array processing**. Commun. Inf. Syst., 05(1):123–130, 2005.

[29] OPPENHEIM, A. V.; SCHAFER, R. W.. **Discrete-Time Signal Processing**. Pearson, 3 edition, 2009.

[30] LIU, C.-L.; VAIDYANATHAN, P. P.. **Super nested arrays: Sparse arrays with less mutual coupling than nested arrays**. In: 2016 IEEE INTERNATIONAL CONFERENCE ON ACOUSTICS, SPEECH AND SIGNAL PROCESSING (ICASSP), p. 2976–2980. IEEE, mar 2016.

[31] FRIEDLANDER, B.; WEISS, A.. **Direction finding in the presence of mutual coupling**. IEEE Transactions on Antennas and Propagation, 39(3):273–284, mar 1991.

[32] SVANTESSON, T.. **Mutual coupling compensation using subspace fitting**. In: PROCEEDINGS OF THE 2000 IEEE SENSOR ARRAY AND MULTICHANNEL SIGNAL PROCESSING WORKSHOP. SAM 2000 (CAT. NO.00EX410), volumen 2000-Janua, p. 494–498. IEEE, 2000.

[33] PAL, P.; VAIDYANATHAN, P. P.. **Coprime sampling and the music algorithm**. In: 2011 DIGITAL SIGNAL PROCESSING AND SIGNAL PROCESSING EDUCATION MEETING (DSP/SPE), volumen 0, p. 289–294. IEEE, jan 2011.

[34] VERTATSCHITSCH, E.; HAYKIN, S.. **Nonredundant arrays**. Proceedings of the IEEE, 74(1):217–217, 1986.

[35] DOLLAS, A.; RANKIN, W. ; MCCRACKEN, D.. **A new algorithm for Golomb ruler derivation and proof of the 19 mark ruler**. IEEE Transactions on Information Theory, 44(1):379–382, 1998.

[36] RUAN, H.; DE LAMARE, R. C.. **Robust Adaptive Beamforming Using a Low-Complexity Shrinkage-Based Mismatch Estimation Algorithm**. IEEE Signal Processing Letters, 21(1):60–64, jan 2014.

[37] FA, R.; DE LAMARE, R. C. ; WANG, L.. **Reduced-Rank STAP Schemes for Airborne Radar Based on Switched Joint Interpolation, Decimation and Filtering Algorithm**. IEEE Transactions on Signal Processing, 58(8):4182–4194, aug 2010.

[38] TUNCER, T. E.; YASAR, T. K. ; FRIEDLANDER, B.. **Direction of arrival estimation for nonuniform linear arrays by using array interpolation**. Radio Science, 42(4):n/a–n/a, aug 2007.

[39] JAFFER, A.. **Maximum likelihood direction finding of stochastic sources: a separable solution**. In: ICASSP-88., INTERNATIONAL CONFERENCE ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING, p. 2893–2896. IEEE, 1988.

[40] LIU, C. L.; VAIDYANATHAN, P. P.. **Remarks on the spatial smoothing step in coarray MUSIC**. IEEE Signal Processing Letters, 22(9):1438–1442, 2015.

[41] BARANIUK, R. G.. **Compressive Sensing**. IEEE Signal Processing Magazine, 24(1):118–120; 124, 2007.

[42] CHARTRAND, R.; BARANIUK, R. G.; ELDAR, Y. C.; FIGUEIREDO, M. A. T. ; TANNER, J.. **Introduction to the Issue on Compressive**

Sensing. IEEE Journal of Selected Topics in Signal Processing, 4(2):241–243, 2010.

[43] YANG, Z.; DE LAMARE, R. C. ; LI, X.. **$L\_1$-Regularized STAP Algorithms With a Generalized Sidelobe Canceler Architecture for Airborne Radar**. IEEE Transactions on Signal Processing, 60(2):674–686, feb 2012.

[44] XU, S.; DE LAMARE, R. C. ; POOR, H. V.. **Distributed Compressed Estimation Based on Compressive Sensing**. IEEE Signal Processing Letters, 22(9):1311–1315, sep 2015.

[45] FRIEDBERG, S.; INSEL, A. ; SPENCE, L.. **Linear Algebra**. Featured Titles for Linear Algebra (Advanced) Series. Pearson Education, 2003.

[46] ELDAR, Y. C.; KUTYNIOK, G.. **Compressed Sensing**. Cambridge University Press, Cambridge, 2012.

[47] MALLAT, S.; ZHIFENG ZHANG. **Matching pursuits with time-frequency dictionaries**. IEEE Transactions on Signal Processing, 41(12):3397–3415, 1993.

[48] DAI, W.; MILENKOVIC, O.. **Subspace pursuit for compressive sensing signal reconstruction**. IEEE Transactions on Information Theory, 55(5):2230–2249, 2009.

[49] BLUMENSATH, T.; DAVIES, M. E.. **Iterative hard thresholding for compressed sensing**. Applied and Computational Harmonic Analysis, 27(3):265–274, 2009.

[50] STOECKLE, C.; MUNIR, J.; MEZGHANI, A. ; NOSSEK, J. A.. **DoA Estimation Performance and Computational Complexity of Subspace- and Compressed Sensing-based Methods**. WSA 2015; 19th International ITG Workshop on Smart Antennas; Proceedings of, p. 1–6, 2015.

[51] GARC, Y. E.. **Randomized Multiple Candidate Iterative Hard Thresholding Algorithm for Direction of Arrival Estimation**. WSA, 1:16–19, 2018.

[52] MALIOUTOV, D.; ÇETIN, M. ; WILLSKY, A. S.. **A sparse signal reconstruction perspective for source localization with sensor arrays**. IEEE Transactions on Signal Processing, 53(8):3010–3022, 2005.

[53] TIBSHIRANI, R.. **Regression shrinkage and selection via the lasso**. Journal of the Royal Statistical Society. Series B (Methodological), 58(1):267–288, 1996.

[54] JIAN WANG; SEOKBEOP KWON ; BYONGHYO SHIM. **Generalized Orthogonal Matching Pursuit**. IEEE Transactions on Signal Processing, 60(12):6202–6216, dec 2012.

[55] RANI, M.; DHOK, S. B. ; DESHMUKH, R. B.. **A Systematic Review of Compressive Sensing: Concepts, Implementations and Applications**. IEEE Access, 6(c):4875–4894, 2018.

[56] DONOHO, D. L.; TSAIG, Y.; DRORI, I. ; STARCK, J.-L.. **Sparse Solution of Underdetermined Systems of Linear Equations by Stagewise Orthogonal Matching Pursuit**. IEEE Transactions on Information Theory, 58(2):1094–1121, feb 2012.

[57] MA, W. K.; HSIEH, T. H. ; CHI, C. Y.. **DOA estimation of quasi-stationary signals via Khatri-Rao subspace**. ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings, p. 2165–2168, 2009.

[58] MA, W.-K.; HSIEH, T.-H.; CHI, C.-Y. ; MEMBER, S.. **DOA Estimation of Quasi-Stationary Signals With Less Sensors Than Sources and Unknown Spatial Noise Covariance : A Khatri − Rao Subspace Approach**. 58(4):2168–2180, 2010.

[59] OBERMEIER, R.; MARTINEZ-LORENZO, J. A.. **Sensing Matrix Design via Mutual Coherence Minimization for Electromagnetic Compressive Imaging Applications**. IEEE Transactions on Computational Imaging, 3(2):217–229, 2017.

[60] DETERME, J. F.; LOUVEAUX, J.; JACQUES, L. ; HORLIN, F.. **Simultaneous Orthogonal Matching Pursuit With Noise Stabilization: Theoretical Analysis**. p. 1–37, jun 2015.