EVALUATION OF CLASSICAL AND SPARSITY-BASED METHODS FOR PARAMETRIC RECOVERY PROBLEMS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

HASAN CAN BAŞKAYA

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

JANUARY 2020

Approval of the thesis:

EVALUATION OF CLASSICAL AND SPARSITY-BASED METHODS FOR PARAMETRIC RECOVERY PROBLEMS

submitted by **HASAN CAN BAŞKAYA** in partial fulfillment of the requirements for the degree of **Master of Science in Electrical and Electronics Engineering Department, Middle East Technical University** by,

Prof. Dr. Halil Kalıpçılar
Dean, Graduate School of **Natural and Applied Sciences** _____

Prof. Dr. İlkay Ulusoy
Head of Department, **Electrical and Electronics Engineering** _____

Assist. Prof. Dr. Figen S. Öktem
Supervisor, **Electrical and Electronics Engineering, METU** _____

Examining Committee Members:

Prof. Dr. Tolga Çiloğlu
Electrical and Electronics Engineering, METU _____

Assist. Prof. Dr. Figen S. Öktem
Electrical and Electronics Engineering, METU _____

Prof. Dr. Umut Orguner
Electrical and Electronics Engineering, METU _____

Assist. Prof. Dr. Elif Vural
Electrical and Electronics Engineering, METU _____

Assist. Prof. Dr. Aykut Koç
Electrical and Electronics Engineering, Bilkent University _____

Date:

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: Hasan Can Başkaya

Signature :

ABSTRACT

EVALUATION OF CLASSICAL AND SPARSITY-BASED METHODS FOR PARAMETRIC RECOVERY PROBLEMS

Başkaya, Hasan Can
M.S., Department of Electrical and Electronics Engineering
Supervisor: Assist. Prof. Dr. Figen S. Öktem

January 2020, 101 pages

Parametric reconstruction problems arise in many areas such as array processing, wireless communication, source separation, and spectroscopy. In a parametric recovery problem, the unknown model parameters in each superimposed signal are estimated from noisy observations. Classical methods perform the recovery over directly on the continuous-valued parameter space by solving a nonlinear inverse problem. Recently sparsity-based methods have also been applied to parametric recovery problems. These methods discretize the parameter space to form a dictionary whose atoms correspond to candidate parameter values, represent the data as a linear combination of small number of dictionary atoms, and then solve the resulting linear inverse problem. These sparsity-based methods can be classified into three categories, namely, on-grid, off-grid and gridless sparse methods. On-grid methods require that the true parameter values lie on a set of fixed grid points. Off-grid methods also use a grid, but the recovered parameter values are allowed to be out of the grid points. On the other hand, gridless methods do not require a grid and they work directly in the continuous-valued parameter space. In this thesis, we first review the classical and sparsity-based

methods developed for parametric recovery problems with single or multiple measurement vectors. We then analyze and evaluate these methods in the direction-of-arrival and parameterized source separation problems.

Keywords: parametric recovery, inverse problem, sparsity, block-sparsity, joint sparsity, single measurement vector, multiple measurement vector, direction-of-arrival, parameterized source separation

ÖZ

PARAMETRİK GERİÇATIM PROBLEMLERİNDE KLASİK VE SEYREKLİK TABANLI YÖNTEMLERİN DEĞERLENDİRİLMESİ

Başkaya, Hasan Can
Yüksek Lisans, Elektrik ve Elektronik Mühendisliği Bölümü
Tez Yöneticisi: Dr. Öğr. Üyesi. Figen S. Öktem

Ocak 2020 , 101 sayfa

Parametrik geriçatım problemleri dizilim işleme, telsiz iletim, spektroskopi ve kaynak ayırma gibi alanlarda ortaya çıkmaktadır. Bir parametrik geriçatım probleminde üst üste binmiş her bir sinyalin bilinmeyen model parametreleri gürültülü ölçümlerden kestirilir. Klasik yöntemler sürekli değerli parametre uzayında doğrusal olmayan bir ters problemi doğrudan çözerek geriçatım yapar. Son zamanlarda, seyreklik tabanlı yöntemler de parametrik geriçatım problemlerine uygulanmıştır. Bu yöntemler parametre uzayını bölerek, atomları aday parametre değerlerine tekabül eden bir sözlük oluştururlar. Sonuç olarak, veri az sayıda sözlük atomlarının doğrusal katışımı olarak temsil edilir ve bu yöntemler ortaya çıkan doğrusal ters problemi çözerler. Bu seyreklik tabanlı yöntemler ızgara üzerinde, ızgara dışında ve ızgarasız seyrek yöntemler olarak üç kategoriye ayrılır. Izgara üzerinde seyrek yöntemler gerçek parametre değerlerinin sabit ızgara noktaları üzerinde olmasını gerektirir. Izgara dışında olan yöntemler de ızgara kullanır, ancak geriçatılan parametre değerlerinin ızgara dışındaki noktalarda olmasına izin verilir. Diğer yandan, ızgarasız yöntemler bir ızgara gerektirmez ve doğrudan sürekli değerli parametre uzayında çalışır. Bu tezde ilk olarak

tekli ölçüm vektörü veya çoklu ölçüm vektörleri ile parametrik geriçatım problemleri için klasik ve seyreklik tabanlı yöntemleri inceledik. Daha sonra bu yöntemleri geliş yönü ve parametreli hale getirilebilir kaynak ayırma problemleri için analiz ettik.

Anahtar Kelimeler: parametrik geriçatım, ters problem, seyreklik, blok seyreklik, ortak seyreklik, tekli ölçüm vektörü, çoklu ölçüm vektörü, geliş yönü, parametreli hale getirilebilir kaynak ayırma

To my family...

ACKNOWLEDGMENTS

First and foremost, I would like to express my deep gratitude and appreciation to my supervisor Assist. Prof. Dr. Figen S. Öktem for her guidance, patience, continuous support and encouragement that made this study possible. I learned a lot from her about research and writing. I thank her for teaching me looking from a scientific perspective and being patient about the results of the experiments.

I would like to express my gratitude to Prof. Dr. Tolga Çiloğlu, Prof. Dr. Umut Orguner, Asst. Prof. Dr. Elif Vural, and Asst. Prof. Dr. Aykut Koç for showing interest in my studies and accepting to review my thesis. I also would like to express my gratitude to ASELSAN A.Ş. for providing the opportunity to fulfil my study.

I would like to give my thanks to my family for their understanding, love and support throughout my life. Finally, I am specially thankful to my wife Didem for her love, emotional support, and being my best friend.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

2D	2 Dimensional
3D	3 Dimensional
ADMM	Alternating Direction Method of Multipliers
ALS	Alternating Least-Squares
ANM	Atomic Norm Minimization
BP	Basis Pursuit
BPDN	Basis Pursuit Denoising
CS	Compressed Sensing
CoSaMP	Compressed Sensing Matching Pursuit
DOA	Direction-of-Arrival
ESPRIT	Estimation of Signal Parameters Via Rotational Invariance Techniques
EPR	Electron Paramagnetic Imaging
FOCUSS	Focal Underdetermined System Solver
LS	Least-Squares
MLE	Maximum Likelihood Estimation
MMV	Multiple Measurement Vector
MP	Matching Pursuit
MRI	Magnetic Resonance Imaging
MUSIC	Multiple Signal Classification
NLS	Nonlinear Least-Squares
NSP	Null Space Property
OGSBI	Off-Grid Sparse Bayesian Inference
OMP	Orthogonal Matching Pursuit

RIP	Restricted Isometry Property
RMSE	Root-Mean-Square-Error
SDP	Semi Definite Programming
SMV	Single Measurement Vector
SNR	Signal-to-Noise Ratio
SOC	Second Order Cone
SR	Success Rate
SVD	Singular Value Decomposition
ULA	Uniform Linear Array
WLLN	Weak Law of Large Numbers

CHAPTER 1

INTRODUCTION

Parametric reconstruction problems arise in many areas, such as array processing, wireless communication, source separation, and spectroscopy. In a parametric recovery problem, the unknown model parameters in each superimposed signal are estimated from noisy observations. That is in this inverse problem, a parametric model that consists of superimposed signals with one or more unknown parameters is fitted to the noisy measurements.

Different types of superimposed signals are encountered in the parametric reconstruction problems depending on the application. The most common one is a complex exponential, which arises in the narrow-band direction-of-arrival problem in sensor array processing [1], attenuation in MRI [2] and line spectral estimation [3]. Moreover, decaying complex exponentials are used to model radar scattering [4]. In spectroscopy [5], [6], [7], [8] and parameterized source separation [9], superimposed Gaussian spectral lines are used to model the spectrum. Lorentzian line-shapes are also used to model the parametric lines in some imaging applications [10]. Other examples include Ricker wavelets used in the biomedical signal processing of intracardiac electrograms [11], and Gamma functions that model the electrical pulse for activity estimation in γ-spectrometry [12].

For the solution of parametric recovery problems, several approaches have been developed, which can be categorized as classical methods and sparsity-based methods. Classical methods perform the recovery over directly on the continuous-valued parameter space by solving a nonlinear inverse problem. Recently sparsity-based methods have also been applied to parametric recovery problems. These methods discretize the parameter space to form a dictionary whose atoms correspond to can-

didate parameter values, represent the data as a linear combination of a small number of dictionary atoms, and then solve the resulting linear inverse problem.

These sparsity-based methods can be classified into three categories, namely, on-grid, off-grid, and gridless sparse methods. On-grid methods require that the true parameter values lie on a set of fixed grid points. Off-grid methods also use a grid, but the recovered parameter values are allowed to be out of the grid points. On the other hand, gridless methods do not require a grid, and they work directly in the continuous-valued parameter space.

In this thesis, we first review the classical and sparsity-based methods developed for parametric recovery problems with single or multiple measurement vectors. We then analyze and evaluate these methods in the direction-of-arrival and parameterized source separation problems.

The organization of this thesis is as follows. The classical methods are reviewed in Chapter 2, and sparsity-based methods are reviewed in Chapter 3. In Chapter 4, these methods are compared for the two different parametric recovery problems, namely direction-of-arrival estimation, and parameterized source separation. Chapter 5 provides conclusions and discussions of the methods.

CHAPTER 2

CLASSICAL METHODS FOR PARAMETRIC RECONSTRUCTION

2.1 Classical Forward Problem

Classical methods for parametric reconstruction use the nonlinear forward model. In other words, the dependence of measurements on the parameters to be reconstructed is nonlinear.

There are two kinds of classical forward problem model of parametric reconstruction in terms of the number of measurements. These are single measurement vector and multiple measurement vectors parametric models.

2.1.1 Single Measurement Vector (SMV) Case

The single measurement vector parametric forward problem has a sequence of measurement samples that can be expressed with a single vector. Single measurement vector (SMV) refers to this single vector. This problem has the basic form among the parametric forward problems because the number of measurement is one. We denote the measurement as a superposition of parametric source signals. Each source signal is completely represented with a few parameters. The mathematical model for the measurement samples are given by

$$y_i = \sum_{n=1}^{N} s_n \phi(i, \boldsymbol{\theta}_n) + w_i \ , \ i = 1, 2, ..., T \tag{2.1}$$

where y_i is the i$^{\text{th}}$ measurement sample, N is the number of source signals, i.e. the model order, s_n is the coefficient of the n$^{\text{th}}$ source signal, $\phi(i, \boldsymbol{\theta}_n)$ is the value at the i$^{\text{th}}$ sample of the n$^{\text{th}}$ source signal which has parameter vector $\boldsymbol{\theta}_n$ and w_i is the noise

value at the i[th] sample. The parameter vector $\boldsymbol{\theta}_n$ consists of P number of parameters, i.e. $\boldsymbol{\theta}_n = [\theta_{n_1}\ \theta_{n_2}\ ...\ \theta_{n_P}]^T$.

The sequence $\{\phi(i, \boldsymbol{\theta}_n)\}_{i=1}^T$ represents the parametric function of the source signal having the parameter vector $\boldsymbol{\theta}_n$. The parametric function of the source signals $\{\phi(i, \boldsymbol{\theta}_n)\}_{i=1}^T$ for all i and the measurement $\{y_i\}_{i=1}^T$ are assumed to be known. The unknowns are parameter vectors, $\{\boldsymbol{\theta}_n\}_{n=1}^N$ and coefficients, $\{s_n\}_{n=1}^N$. The model order, N is known in some problems whereas it is unknown in the other problems. There is a nonlinear dependency between the parameters, $\boldsymbol{\theta}_n$ and the measurements $\{y_i\}$. Then, we can model Equation (2.1) as a nonlinear system using matrix-vector multiplication.

Let $\boldsymbol{y}$ and $\boldsymbol{w}$ be vectors consist of the measurement samples y_i and noise sample w_i as their i[th] element, respectively. Also, the coefficients are gathered into a N-point vector, $\boldsymbol{s}$. The parametric function of source signals $\{\phi(i, \boldsymbol{\theta}_n)\}_{i=1}^T$ can be written as a vector which is $\boldsymbol{a}(\boldsymbol{\theta_n}) = [\phi(1, \boldsymbol{\theta_n}), ..., \phi(i, \boldsymbol{\theta_n}), ..., \phi(T, \boldsymbol{\theta_n})]^T$. Then, the nonlinear system model of Equation (2.1) is given by

$$\boldsymbol{y} = \boldsymbol{A}(\boldsymbol{\Theta})\boldsymbol{s} + \boldsymbol{w} \tag{2.2}$$

where $\boldsymbol{s} = [s_1, .., s_N]^T \in \mathbb{C}^N$, $\boldsymbol{y} = [y_1, .., y_T]^T \in \mathbb{C}^T$, $\boldsymbol{w} = [w_1, .., w_T]^T \in \mathbb{C}^T$, $\boldsymbol{\Theta} = [\boldsymbol{\theta_1}, .., \boldsymbol{\theta_N}] \in \mathbb{R}^{P \times N}$, $\boldsymbol{\theta_n} = [\theta_{n_1}\ \theta_{n_2}\ ...\ \theta_{n_P}]^T$ and $\boldsymbol{A}(\boldsymbol{\Theta}) = [\boldsymbol{a}(\boldsymbol{\theta_1}), .., \boldsymbol{a}(\boldsymbol{\theta_N})]$ $\in \mathbb{C}^{T \times N}$. Note that instead of the above complex spaces, the real spaces can be used as a special case in some problems.

A single measurement vector case is generally utilized as a simple version of the multiple measurement vector case. In real-life problems, we generally encounter multiple measurement vectors scenarios, which will be defined in the next section. However, a single measurement vector case is also used as the most straightforward setting for algorithm development [13], [14], and for the problems where using the multiple measurements is not possible [11].

2.1.2 Multiple Measurement Vector (MMV) Case

In case of multiple measurement vectors, different single measurement vectors are combined. These different measurements can be taken in various ways. For example,

the system in a noisy environment is observed in consecutive time instants. In this case, it can be assumed that the values of the parameters of the parametric functions do not change over the measurements because there is often a short acquisition time between the measurements which holds the value of parameters unchanged over time. In some other cases the values of parameters change slowly with different measurements. For example, in an astrophysical imaging problem, a measurement represents one pixel of the scene. There are small changes between the parameters that characterizes the data in neighboring pixels because the images in this problem are smooth. However, we first introduce the general parametric model for MMV in which the parameter values change with different measurements:

$$y_{i,l} = \sum_{n=1}^{N} s_{n,l}\phi(i, \boldsymbol{\theta}_{n,l}) + w_{i,l} \ , \ i = 1, 2, ..., T \text{ and } l = 1, 2, ..., L \tag{2.3}$$

where $y_{i,l}$ is the i^{th} measurement sample in the l^{th} measurement, N is the number of source signals, i.e. the model order, $s_{n,l}$ is the coefficient of the n^{th} source signal in the l^{th} measurement, $\phi(i, \boldsymbol{\theta}_{n,l})$ is the value at the i^{th} sample of the n^{th} source signal in l^{th} measurement which has parameter vector $\boldsymbol{\theta}_{n,l}$ and $w_{i,l}$ is the noise value at the i^{th} sample in the l^{th} measurement.

The sequence $\{\phi(i, \boldsymbol{\theta}_{n,l})\}_{i=1}^{T}$ represents the parametric function of the source signal having the parameter vector $\boldsymbol{\theta}_{n,l}$. The parametric functions of the source signals $\{\phi(i, \boldsymbol{\theta}_{n,l})\}_{i=1}^{T}$ for all i and l and $\{y_{i,l}\}_{i=1}^{T}$ are assumed to be known. The unknowns are parameter vectors, $\{\boldsymbol{\theta_{n,l}}\}$ and coefficients, $\{s_{n,l}\}$ for all n and l. The model order, N, is known a priori in some applications. On the other hand, it can be unknown in the other applications.. There is a nonlinear dependency between the parameters, $\boldsymbol{\theta_{n,l}}$ and l^{th} measurement $\{y_l(i)\}_{i=1}^{T}$ as in SMV case.

The parametric model in Equation (2.3) can be written in a matrix-vector multiplication form:

$$\boldsymbol{y}_l = \boldsymbol{A}(\boldsymbol{\Theta}_l)\boldsymbol{s}_l + \boldsymbol{w}_l \ , \ l = 1, 2, ..., L \tag{2.4}$$

where $\boldsymbol{y}_l$ is the measurement vector, $\boldsymbol{A}(\boldsymbol{\Theta}_l)$ is the matrix that consists of parametric functions, $\boldsymbol{s}_l$ is the vector of coefficients of the superimposed source signals and $\boldsymbol{w}_l$ is the noise vector in l^{th} measurement. The measurement vectors, coefficient vectors

and noise vectors cannot be written in matrix form because there is no common $\boldsymbol{A}(\boldsymbol{\Theta})$ matrix for different measurements.

As a special case of MMV, the parameter vector $\boldsymbol{\theta}_{n,l}$ is the same for different measurements, i.e. $\boldsymbol{\theta}_{n,l} = \boldsymbol{\theta}_n$ in some problems. It is assumed that the only change in the system between the measurements is the noise in this special case. Then, the model in Equation (2.4) reduces to the following model:

$$\boldsymbol{y}_l = \boldsymbol{A}(\boldsymbol{\Theta})\boldsymbol{s}_l + \boldsymbol{w}_l \ , \ l = 1, 2, ..., L \tag{2.5}$$

For this special case, each measurement can be expressed using the common $\boldsymbol{A}(\boldsymbol{\Theta})$ matrix. Then we can write Equation (2.5) as a nonlinear system using matrix multiplications in a compact form.

We define $\boldsymbol{Y}$ and $\boldsymbol{N}$ as $T \times L$ matrices consist of the measurement sample $y_{i,l}$ and noise sample $n_{i,l}$ in their i^{th} row and the l^{th} column, respectively. Also, we define a $N \times L$ matrix, $\boldsymbol{S}$, consist of the coefficient $s_{n,l}$ in its n^{th} row and l^{th} column. If we write the parametric function of source signals $\{\phi(i, \boldsymbol{\theta}_k)\}_{i=1}^{W}$ as a vector which is $\boldsymbol{a}(\boldsymbol{\theta_n}) = [\phi(1, \boldsymbol{\theta_n}), ..., \phi(i, \boldsymbol{\theta_n}), ..., \phi(T, \boldsymbol{\theta_n})]^T$ then, the nonlinear system model is given by

$$\boldsymbol{Y} = \boldsymbol{A}(\boldsymbol{\Theta})\boldsymbol{S} + \boldsymbol{W} \tag{2.6}$$

where $\boldsymbol{Y} = [\boldsymbol{y}_1, .., \boldsymbol{y}_L] \in \mathbb{C}^{T \times L}$ with $\boldsymbol{y}_l = [y_{1,l}, .., y_{T,l}]^T$, $\boldsymbol{A}(\boldsymbol{\Theta}) = [\boldsymbol{a}(\boldsymbol{\theta}_1), .., \boldsymbol{a}(\boldsymbol{\theta}_N)] \in \mathbb{C}^{T \times N}$ with $\boldsymbol{\Theta} = [\boldsymbol{\theta}_1, .., \boldsymbol{\theta}_N] \in \mathbb{R}^{P \times N}$, $\boldsymbol{S} = [\boldsymbol{s}_1, .., \boldsymbol{s}_L] \in \mathbb{C}^{N \times L}$ with $\boldsymbol{s}_l = [s_{1,l}, .., s_{K,l}]^T$, and $\boldsymbol{W} = [\boldsymbol{w}_1, .., \boldsymbol{w}_L] \in \mathbb{C}^{T \times L}$ with $\boldsymbol{w}_l = [n_{1,l}, .., n_{T,l}]^T$.

If the redundancy between the measurement vectors is exploited, the use of the MMV model possibly increases the performance of the solution method.

The parametric reconstruction problems, especially with multiple measurement case, are widely encountered in signal processing. The most common variant of multiple measurement vectors case is the direction-of-arrival estimation problem, which is encountered in many applications such as sonar, radar, wireless communications. Here, exponential signals are used as parametric signals, which will be discussed in the following sections. In this problem, the same system is measured in consecutive times, and it is assumed that the parameters are the same for different measurements.

2.2 Applications

Parametric reconstruction problems are encountered in many areas, including array processing, biomedical signal processing, astrophysical imaging, and spectroscopy. Direction-of-arrival estimation is one of the well-known parametric reconstruction problems that are encountered in sonar, radar, and seismic signal processing.

Parametric reconstruction problems are also encountered in biomedical signal processing. In [2], magnetic resonance imaging (MRI) signals are investigated. The received MRI signal consists of the additive complex exponentials. It should be noted that the signal of exponential form, which is similar to the direction-of-arrival estimation problem is observed in MRI as well. Moreover, biomedical signal processing of intracardiac electrograms is one of the applications of parametric reconstruction problems. Ricker wavelets are used as parametric functions in this application [11]

Astrophysical imaging is another application area of the parametric reconstruction problems. In instantaneous multi-spectral imaging, we again encounter the measurements, which are the superposition of parametric signals. Here, instead of exponential signals, we observe the superposition of Gaussian signals [5]. As an output, there are measurements of a computational 3D multi-spectral imaging system in which 2D detectors are used. The 3D imaging system consists of two spatial dimensions and one spectral dimension. Note that the spectrum is composed of discrete spectral lines having Gaussian shape [5]. Therefore, each pixel in the spatial dimensions consists of a Gaussian spectral line. In the measurements, measured images are expressed as the superposition of dispersed spectral lines from different spatial positions on the scene [5]. Regarding 2D measurements, each column of the measurements only includes the superposition of the Gaussian spectral lines coming from neighboring pixels in the same column. Hence, we express each column of the measured images with different dispersion rates as an MMV parametric reconstruction problem.

The parameterized source separation problem deals with the estimation of emission lines in spectroscopic signals. These emission lines are modeled by a parametric waveform such as Gaussian and Lorentzian [9]. One of the application areas of parameterized source separation is photoelectron spectroscopy. It is an imaging method

that determines the atomic structure of chemical objects. Gaussian spectral lines are also used to model the spectrum of the photoelectron data [9].

We now detailed the two problems, direction-of-arrival and parameterized source separation, because we will apply the classical and sparsity-based methods to these problems.

2.2.1 Direction-of-Arrival Estimation

Direction-of-Arrival (DOA) estimation is a particular case of the frequency estimation problem. In the frequency estimation problem, the aim is to estimate the frequencies of superimposed complex exponential signals from noisy data.

In this problem, the arrival angles of a target or multiple targets are estimated using a sensor array. The arriving waves may represent acoustic, electromagnetic, or seismic signals. The set of T sensors are excited with these waves arriving from the sources in different directions. A few crucial assumptions should be made to model the DOA estimation problem efficiently.

- Assumed that the sources are in the far-field so that the angle of arrival is approximately the same to each sensor of the array. In other words, if the sources are in the far-field, then the arriving waves behave as plane waves. This plane wave assumption holds almost all applications of DOA estimation except for the localization of speech sources in small places. [15]

- Assumed that the arriving waves are narrow-band. Suppose that the signal is band-limited with bandwidth B_s, and the maximum travel time of the wave between two sensors is Δt. If $B_s \times \Delta t << 1$, then the arriving signal is a narrow-band signal. The signal waveforms are the same for all sensors when the narrow-band assumption holds. [15]

- Assumed that the sources and sensor array are in the same plane. Consequently, the elevation angle of the sources is equal to 90°. Direction-of-arrival estimation is performed only in the azimuth plane.

The arrangement of the sensor array is an essential topic for DOA estimation. However, the most popular and simple arrangement, uniform linear array, is used to avoid deviation from the focus of the thesis. The sensors in a uniform linear array are on a straight line, and the distance between consecutive sensors are the same and equal to Δ.

Considering the far field sources and narrow-band signals impinging upon uniform linear array, the signal observed by the i$^{\text{th}}$ sensor is given by

$$y_i = \sum_{n=1}^{N} s_n \exp\{j(i-1)\frac{2\pi}{\lambda}\Delta sin\theta_n\} + w_i \ , \ i = 1, 2, ..., T \tag{2.7}$$

where s_n denotes the coefficient of n$^{\text{th}}$ signal waveform and θ_k is the angle of arrival of that signal.

In a standard uniform linear array, the distance between consecutive sensors is half a wavelength, $\Delta = \lambda/2$. It should be noted that the distance larger than half a wavelength causes spatial aliasing[16]. In this thesis, we use the standard uniform linear array. Consequently, Equation (2.7) is represented as follows:

$$y_i = \sum_{n=1}^{N} s_n \exp\{j(i-1)\pi sin\theta_n\} + w_i \ , \ i = 1, 2, ..., T \tag{2.8}$$

The unknown parameters in this problem are the number of complex exponentials, N and the angles of arrivals of these signals, $\{\theta_n\}$.

Matrix-vector form of the above model is given by

$$\boldsymbol{y} = \boldsymbol{A}(\boldsymbol{\theta})\boldsymbol{s} + \boldsymbol{w} \tag{2.9}$$

where $\boldsymbol{y} = [y_1, ..., y_i, ..., y_T]^T$, $\boldsymbol{A}(\boldsymbol{\theta}) = [\boldsymbol{a}(\theta_1), ..., \boldsymbol{a}(\theta_n), ..., \boldsymbol{a}(\theta_N)]$ with $\boldsymbol{\theta} = [\theta_1, .., \theta_n, ..., \theta_N]$ and $\boldsymbol{a}(\theta_n) = [1, e^{j\pi sin(\theta_n)}, ..., e^{j\pi(T-1)(sin(\theta_n)}]^T$, $\boldsymbol{s} = [s_1, s_2, ..., s_N]^T$, and $\boldsymbol{n} = [w_1, ..., w_i, ..., w_T]^T$. In array processing literature, the vector $\boldsymbol{a}(\theta_n)$ is called as steering vector and the matrix $\boldsymbol{A}(\boldsymbol{\theta})$ is called as steering matrix or array manifold [15]. Note that this forward problem is a special case of the general parametric forward model in Equation (2.2) because the parameter vector $\boldsymbol{\theta}_n$ in Equation (2.2) reduces to the single parameter θ_n which represents the arrival angle in this problem.

In many DOA estimation applications, multiple measurement vectors are available. In array processing literature, multiple measurement vectors that are taken in different

times are referred to as multiple *snapshots*. These multiple snapshots are taken in consecutive time instants. In a multiple snapshots DOA estimation problem, it is mostly assumed that the arrival angles of the sources are the same for each snapshot. In this case, the model in Equation (2.9) replaces with the following model:

$$\boldsymbol{Y} = \boldsymbol{A}(\boldsymbol{\theta})\boldsymbol{S} + \boldsymbol{W} \tag{2.10}$$

where $\boldsymbol{Y} = [\boldsymbol{y_1}, \boldsymbol{y_2}, ..., \boldsymbol{y_L}]$, $\boldsymbol{A}(\boldsymbol{\theta})$ has exactly the same form in Equation (2.9), $\boldsymbol{S} = [\boldsymbol{s_1}, \boldsymbol{s_2}, ..., \boldsymbol{s_L}]$, and $\boldsymbol{W} = [\boldsymbol{w_1}, \boldsymbol{w_2}, ..., \boldsymbol{w_L}]$. The vectors $\boldsymbol{s_l}$, $\boldsymbol{y_l}$ and $\boldsymbol{w_l}$ can be considered the coefficient, measurement and noise vector of the single snapshot case, respectively. If the assumption of the same arrival angles for each snapshot does not hold, the compact form in Equation (2.10) cannot be written because there would not be a common $\boldsymbol{A}(\boldsymbol{\theta})$ matrix for different snapshots.

2.2.2 Parameterized Source Separation

Parameterized source separation arises in various signal processing areas such as time-resolved photoelectron spectroscopy [9], instantaneous spectral imaging [5] and the study of galaxy kinematics [17]. In this problem, the spectrum is composed of spectral emission lines that can be modeled as parametric signals. Each parameter of the spectral emission lines identifies a physical phenomenon. For example, the delay parameter of each spectral emission line in the spectrum identifies the internal gas motions and redshift in galaxy kinematics [17]. The parametric signals can be modeled as Gaussian [17] or Lorentzian [9] functions. The aim is to estimate the intensities, delays, and widths of Gaussian or Lorentzian functions in the spectrum.

Consider L number of spectrum measurements, each of them composed of N parameterized Gaussian sources. Each spectrum can be written as a noisy sum of the Gaussian sources:

$$y_{i,l} = \sum_{n=1}^{N} s_{n,l}\phi(i, \boldsymbol{\theta}_{n,l}) + w_{i,l} \ , \ i = 1, 2, ..., T \text{ and } l = 1, 2, ..., L \tag{2.11}$$

where $y_{i,l}$ is the measured intensity, $s_{n,l}$ is the amplitude of the n^{th} Gaussian source, and $\phi(i, \boldsymbol{\theta}_{n,l})$ denotes the i^{th} sample of the Gaussian source in the l^{th} spectrum. The parameter vector $\boldsymbol{\theta}_n$ consists of two parameters, i.e. $\boldsymbol{\theta}_n = [\theta_{n_1} \ \theta_{n_2}]^T$. θ_{n_1} and θ_{n_2}

represent the delay parameter and the width parameter of n^{th} Gaussian source, respectively. The amplitudes of the sources are nonnegative because the sources are Gaussian emission lines. $w_{i,l}$ is the i^{th} sample of the noise sequence in the l^{th} spectrum. Figure 2.1 shows a single Gaussian source in the spectrum and its parameters.

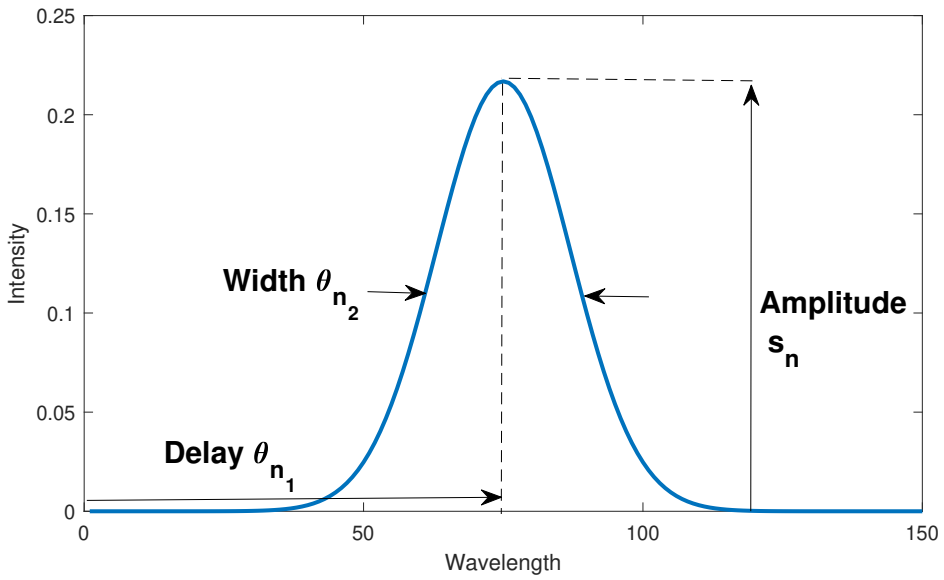

Figure 2.1: Sample Gaussian source and its parameters

Matrix-vector form of the model in Equation (2.11) for l^{th} spectrum is given by

$$\boldsymbol{y}_l = \boldsymbol{A}(\boldsymbol{\Theta}_l)\boldsymbol{s}_l + \boldsymbol{n}_l \tag{2.12}$$

where $\boldsymbol{y}_l = [y_{1,l}, ..., y_{i,l}, ..., y_{T,l}]^T$, $\boldsymbol{A}(\boldsymbol{\Theta}_l) = [\boldsymbol{a}(\boldsymbol{\theta}_1), ..., \boldsymbol{a}(\boldsymbol{\theta}_n), ..., \boldsymbol{a}(\boldsymbol{\theta}_N)]$, $\boldsymbol{s}_l = [s_{1,1}, s_{2,1}, ..., s_{N,1}]^T$, and $\boldsymbol{w}_l = [w_{1,l}, ..., w_{i,l}, ..., w_{T,l}]^T$.

Note that Equation (2.11) is the most general form where values of parameters of the sources change with different spectrum measurements. However, parameters can be considered as being constant over different spectrum measurements in special cases. For example, if each spectrum represents the same single pixel value at different time instants and the measurements are taken in a short time, it can be assumed that the sources are the same for all spectra. In this case, the above model reduces to the

model where all parameters of the sources have constant value over spectra:

$$y_{i,l} = \sum_{n=1}^{N} s_{n,l}\phi(i, \boldsymbol{\theta}_n) + w_{i,l} \ , \ i = 1, 2, ..., T \text{ and } l = 1, 2, ..., L \tag{2.13}$$

Matrix form of the model in Equation (2.13) is given by

$$\boldsymbol{Y} = \boldsymbol{A}(\boldsymbol{\Theta})\boldsymbol{S} + \boldsymbol{N} \tag{2.14}$$

where each column of $\boldsymbol{Y}$, $\boldsymbol{S}$ and $\boldsymbol{N}$ is the measurement, amplitude and noise vector of the corresponding spectrum. The matrix $\boldsymbol{A}(\boldsymbol{\Theta})$ is common for all spectrum measurements.

In the parameterized source separation problem, the aim is to estimate the amplitudes, delays, and widths of the Gaussian sources with the model order, K.

In parameterized source separation problems, there are multiple measurement vectors available. For example, the measurements that are taken in a short time from the same system form the MMV in photoelectron spectroscopy applications [9] and the measurement that are taken from the neighboring image pixels result in the MMV in galaxy kinematics applications [17].

2.3 Nonlinear Least-Squares Method

The parametric reconstruction problems are nonlinear because of the nonlinear dependence between the parameters and the parametric functions. The most common and frequently used model-based approach in signal processing is the nonlinear least-squares (NLS) technique. It requires statistical assumptions for data generation. In this section, we will briefly review this approach, discuss their relative merits. Even though signal models are deterministic and unknown, noise is modeled as stationary white Gaussian noise. With this assumption, the least-squares solution is equivalent to the maximum likelihood estimator (MLE). The term deterministic maximum likelihood is also used because the signal models are admitted as deterministic and unknown.

First,consider the SMV parametric forward model:

$$\boldsymbol{y} = \boldsymbol{A}(\boldsymbol{\Theta})\boldsymbol{s} + \boldsymbol{w} \tag{2.15}$$

The least-squares estimator minimizes the following function [18]:

$$\min_{\boldsymbol{s},\boldsymbol{\Theta}} \|\boldsymbol{y} - \boldsymbol{A}(\boldsymbol{\Theta})\boldsymbol{s}\|_2^2 \tag{2.16}$$

Consider a fixed value of $\boldsymbol{\Theta}$, then the above objective can be considered as a separable nonlinear least-squares problem [19]. Therefore, the minimization with respect to the coefficient vector $\boldsymbol{s}$ can be analytically solved because the measurement vector $\boldsymbol{y}$ is a linear function of the coefficient vector $\boldsymbol{s}$. The solution is given by

$$\hat{\boldsymbol{s}}(\boldsymbol{\Theta}) = ((\boldsymbol{A}(\boldsymbol{\Theta}))^H \boldsymbol{A}(\boldsymbol{\Theta}))^{-1} (\boldsymbol{A}(\boldsymbol{\Theta}))^H \boldsymbol{y} = \boldsymbol{A}(\boldsymbol{\Theta})^\dagger \boldsymbol{y} \tag{2.17}$$

where $\boldsymbol{A}(\boldsymbol{\Theta})^\dagger$ is the Moore-Penrose pseudo-inverse of $\boldsymbol{A}(\boldsymbol{\Theta})$. Substitute this into Equation (2.16), we obtain the following equation:

$$\min_{\boldsymbol{\Theta}} \|\boldsymbol{y} - \boldsymbol{A}(\boldsymbol{\Theta})\boldsymbol{A}(\boldsymbol{\Theta})^\dagger \boldsymbol{y}\|_2^2 = \min_{\boldsymbol{\Theta}} \|\boldsymbol{P}^\perp_{\mathcal{R}(\boldsymbol{A}(\boldsymbol{\Theta}))} \boldsymbol{y}\|_2^2 \tag{2.18}$$

where $\boldsymbol{P}^\perp_{\mathcal{R}(\boldsymbol{A}(\boldsymbol{\Theta}))}$ is the orthogonal projector onto the nullspace of $\boldsymbol{A}(\boldsymbol{\Theta})^H$.

Now, consider the special case of MMV parametric forward model where the parameter values are constant over different measurement:

$$\boldsymbol{Y} = \boldsymbol{A}(\boldsymbol{\Theta})\boldsymbol{S} + \boldsymbol{W} \tag{2.19}$$

The least-squares estimator minimizes the following function:

$$\min_{\boldsymbol{S},\boldsymbol{\Theta}} \|\boldsymbol{Y} - \boldsymbol{A}(\boldsymbol{\Theta})\boldsymbol{S}\|_F^2 \tag{2.20}$$

where $\|.\|_F$ is the Frobenius matrix norm. The same procedure with the SMV case is valid for the MMV case. The resultant objective function is similar to that of the SMV case and it is given by

$$\min_{\boldsymbol{\Theta}} \|\boldsymbol{P}^\perp_{\mathcal{R}(\boldsymbol{A}(\boldsymbol{\Theta}))} \boldsymbol{Y}\|_2^2 = \min_{\boldsymbol{\Theta}} \mathrm{Tr}\{\boldsymbol{P}^\perp_{\mathcal{R}(\boldsymbol{A}(\boldsymbol{\Theta}))} \hat{\boldsymbol{R}}\} \tag{2.21}$$

where $\hat{\boldsymbol{R}} = \frac{1}{L}\boldsymbol{Y}\boldsymbol{Y}^H$ and Tr is the trace operator.

Here, a N-dimensional optimization problem should be solved where N is the model order. After obtaining $\boldsymbol{\Theta}$, the signal waveform and the noise variance is directly found by inserting $\boldsymbol{\Theta}$ value. With a good initial guess, the Gauss-Newton method converges to the minimum, usually [20]. However, the computational cost is usually expensive to obtain accurate initial estimates. If initial estimates are weak, the search procedure might be stuck into a local-minimum value.

2.4 Subspace-based Methods

Subspace-based methods estimate the parameters from the sample correlation of the measurements, and they require the model order as prior information. These methods are part of classical methods because they use the classical forward model.

In the following sections, two subspace-based methods that are well-known in the literature, MUSIC, and ESPRIT are discussed. Note that these methods apply only to the parametric reconstruction problems having sinusoidal parameters.

2.4.1 MUSIC

Multiple Signal Classification (MUSIC) method is a high-resolution eigenstructure method that estimates parameters in a parametric reconstruction problem. The method is suggested in [21]. It is specifically associated with one of the parametric reconstruction problems, the direction-of-arrival estimation problem. However, the MUSIC method can be applied to other parametric reconstruction problems having sinusoidal parameters. [22] demonstrates some of the practical applications of the MUSIC method.

The method can be applied only to the special case of the MMV parametric forward model, where the parameter values are constant over different measurement because the method is based on the sample correlation matrix of the measurements. It is necessary to gather a sufficient number of measurements to obtain a sample correlation matrix reliably. This number should be at least larger than $3T$, where T is the number of measurement samples in a measurement [23]. The number of measurements should be much larger if the superimposed signals are closely placed. Let us take the special case of MMV parametric forward model where the parameter values are constant over different measurement:

$$\boldsymbol{Y} = \boldsymbol{A}(\boldsymbol{\Theta})\boldsymbol{S} + \boldsymbol{W} \tag{2.22}$$

Before the mathematical derivations, some assumptions are required:

- Assume that $\boldsymbol{A}(\boldsymbol{\Theta})$ has full column rank.

- Assume that the number of parameters to be estimated is smaller than or equal to number of measurement, i.e. $N \leq T$.
- Assume that the signal components and the additive noise are uncorrelated.

First, consider the sample correlation matrix of $\boldsymbol{Y}$ in Equation (2.22) which is given by

$$\hat{\boldsymbol{R}} = \frac{1}{L}\boldsymbol{Y}\boldsymbol{Y}^H = \frac{1}{L}\sum_{l=1}^{L}\boldsymbol{y}_l\boldsymbol{y}_l^H \tag{2.23}$$

where $\boldsymbol{y}_l$ is the l^{th} column of the measurement matrix $\boldsymbol{Y}$. From Equation (2.22), Equation (2.23) and above assumptions, we have

$$\hat{\boldsymbol{R}} = \boldsymbol{A}(\boldsymbol{\Theta})\left(\frac{1}{L}\sum_{l=1}^{L}\boldsymbol{s}_l\boldsymbol{s}_l^H\right)\boldsymbol{A}^H(\boldsymbol{\Theta}) + \frac{1}{L}\sum_{l=1}^{L}\boldsymbol{w}_l\boldsymbol{w}_l^H \tag{2.24}$$

where $\boldsymbol{s}_l$ and $\boldsymbol{w}_l$ is the l^{th} column of matrices $\boldsymbol{S}$ and $\boldsymbol{W}$, respectively. From the weak law of large numbers (WLLN), as $L \to \infty$, the sample correlation matrix converges to

$$\hat{\boldsymbol{R}} = \boldsymbol{A}(\boldsymbol{\Theta})\hat{\boldsymbol{R}}_{\boldsymbol{S}}\boldsymbol{A}^H(\boldsymbol{\Theta}) + \sigma^2\boldsymbol{I} \tag{2.25}$$

where $\hat{\boldsymbol{R}}_{\boldsymbol{S}}$ is the sample correlation matrix of the coefficient matrix, $\boldsymbol{S}$ and σ^2 is the noise variance. Let's take the eigenvalue decomposition of the sample correlation matrix $\hat{\boldsymbol{R}}$:

$$\hat{\boldsymbol{R}} = \begin{bmatrix}\hat{\boldsymbol{U}}_s\hat{\boldsymbol{U}}_w\end{bmatrix}\begin{bmatrix}\sigma_1^2+\sigma^2 & & & & & & \\ & \ddots & & & & & \\ & & \sigma_N^2+\sigma^2 & & & & \\ & & & \sigma^2 & & & \\ & & & & \ddots & & \\ & & & & & \sigma^2 & \end{bmatrix}\begin{bmatrix}\hat{\boldsymbol{U}}_s^H \\ \hat{\boldsymbol{U}}_w^H\end{bmatrix} \tag{2.26}$$

Ideally, the first N left singular vectors span the signal subspace and the remaining ones span the noise subspace. We know that $\mathcal{R}(\boldsymbol{U_s}) = \mathcal{R}(\boldsymbol{U_w})^{\perp}$. Also, range spaces of the matrix $\boldsymbol{A}(\boldsymbol{\Theta})$ and $\boldsymbol{U_s}$ are equal to each other. As a result, $\mathcal{R}(\boldsymbol{A}(\boldsymbol{\Theta})) = \mathcal{R}(\boldsymbol{U_w})^{\perp}$. Then, the following holds:

$$\boldsymbol{U_w}^H\boldsymbol{a}(\theta) = 0 \Leftrightarrow \theta = \theta_n \tag{2.27}$$

We define a function of θ as follows:

$$g(\theta) = \frac{1}{||\hat{\boldsymbol{U}}_{\boldsymbol{w}}{}^{H}\boldsymbol{a}(\theta)||} \tag{2.28}$$

If we plot the function $g(\theta)$ over all parameter space, the function will have peaks at the true parameter points. This function is called as the "MUSIC Spectrum". The peak points of the MUSIC spectrum give us the estimated parameter values. Note that, $\hat{\boldsymbol{U}}_{\boldsymbol{w}} \to \boldsymbol{U}_{\boldsymbol{w}}$ as $L \to \infty$, the MUSIC spectrum can resolve closely spaced sources as L and SNR increases.

The main computational complexity comes from the decomposition of the sample correlation matrix and the calculation of the MUSIC spectrum. MUSIC method calculates the eigenvalue decomposition of the sample correlation matrix, which has the computational complexity of $\mathcal{O}(T^3)$. Also, the calculation of the MUSIC spectrum requires a one-dimensional search over K points, where K is the number of grid points in the function $g(\theta)$.

2.4.2 ESPRIT

Estimation of Signal Parameters Via Rotational Invariance Techniques (ESPRIT) method is one of the subspace-based methods that solve the parametric reconstruction problems [24]. Similar to the MUSIC method, it uses the sample correlation matrix of the measurements. Therefore, the ESPRIT method requires a sufficient number of measurements to construct a sample correlation matrix reliably. The MMV parametric model will be used in this method. ESPRIT method is specifically designed for the direction-of-arrival estimation problem, but it can apply to the other parametric reconstruction problems. For example, ESPRIT is used in the estimation of the modes of oscillation in the frequency in a power system [25]. We consider the problem in 2.2.1 with multiple snapshots when explaining the ESPRIT method.

ESPRIT method for sensor array signal processing is applicable to arrays that have shift-invariance property [15]. We use uniform linear array arrangement in this thesis. Uniform linear arrays have shift-invariance property. Then, ESPRIT is applicable to uniform linear arrays [15]. Consider a T-element ULA and divide the array into two sub-arrays. Assume that the first $T-1$ sensors belong to the sub-array Y_1 and the

last $T-1$ sensors belong to the sub-array Y_2. The MMV parametric model assuming each superimposed source signal is expressed with one parameter that is in DOA estimation case is given by

$$\boldsymbol{Y} = \boldsymbol{A}(\boldsymbol{\theta})\boldsymbol{S} + \boldsymbol{W} \tag{2.29}$$

Therefore, the parametric signal model of the sub-arrays is given by

$$\boldsymbol{Y_1} = \boldsymbol{A}(\boldsymbol{\theta})\boldsymbol{S} + \boldsymbol{W_1} \quad \text{and} \quad \boldsymbol{Y_2} = \boldsymbol{A}(\boldsymbol{\theta})\boldsymbol{\Phi}\boldsymbol{S} + \boldsymbol{W_2}$$

$$\text{where} \quad \boldsymbol{\Phi} = \begin{bmatrix} e^{-j2\pi\frac{\Delta}{\lambda}sin(\theta_1)} & & 0 \\ & \ddots & \\ 0 & & e^{-j2\pi\frac{\Delta}{\lambda}sin(\theta_N)} \end{bmatrix}$$

We define a matrix as follows:

$$\boldsymbol{Z} = [\boldsymbol{Y_1}\ \boldsymbol{Y_2}]^T = \boldsymbol{A}_1\boldsymbol{S} + [\boldsymbol{W_1}\ \boldsymbol{W_2}]^T$$

where $\boldsymbol{A}_1 = [\boldsymbol{A}\ \boldsymbol{A\Phi}]^T$. From the weak law of large numbers (WLLN), as $L \to \infty$, the sample correlation matrix of $\boldsymbol{Z}$ converges to

$$\hat{\boldsymbol{R}}_{\boldsymbol{z}} = \boldsymbol{A}(\boldsymbol{\theta})\hat{\boldsymbol{R}}_{\boldsymbol{s}}\boldsymbol{A}^H(\boldsymbol{\theta}) + \sigma^2\boldsymbol{I} \tag{2.30}$$

where $\hat{\boldsymbol{R}}_{\boldsymbol{s}}$ is the sample correlation matrix of the coefficient matrix of source signals and σ^2 is the noise variance. Similar to MUSIC method, we take SVD of $\hat{\boldsymbol{R}}_{\boldsymbol{z}}$ and define the matrix $\hat{\boldsymbol{U}}_{\boldsymbol{s}}$ that consists of the first N left singular vectors. This matrix is related to the matrix $\boldsymbol{A}_1$ with a linear transformation because both matrices span the same signal space, ideally. Therefore, the following holds:

$$\boldsymbol{U_s} = \boldsymbol{A}_1\boldsymbol{\Delta} = \begin{bmatrix} \boldsymbol{A\Delta} \\ \boldsymbol{A\Phi\Delta} \end{bmatrix} = \begin{bmatrix} \boldsymbol{U_1} \\ \boldsymbol{U_2} \end{bmatrix} \tag{2.31}$$

We define a matrix, $\boldsymbol{U} = [\boldsymbol{U_1}\ \boldsymbol{U_2}]$. Then, there exists a matrix, $\boldsymbol{F} = [\boldsymbol{F_1}\ \boldsymbol{F_2}]^T$ that fulfills the following equation:

$$\boldsymbol{UF} = \boldsymbol{U_1F_1} + \boldsymbol{U_2F_2} = \boldsymbol{A\Delta F_1} + \boldsymbol{A\Phi\Delta F_2} = 0 \tag{2.32}$$

This equation implies that

$$\boldsymbol{A\Delta}(-\boldsymbol{F_1}{\boldsymbol{F_2}}^{-1}) = \boldsymbol{A\Phi\Delta} \tag{2.33}$$

By using the matrix $\boldsymbol{\Psi} = -\boldsymbol{F_1}\boldsymbol{F_2}^{-1}$, Equation (2.33) can be written as $\boldsymbol{\Psi} = \boldsymbol{\Delta}^{-1}\boldsymbol{\Phi}\boldsymbol{\Delta}$. From here, we can say that the diagonal elements of matrix $\boldsymbol{\Phi}$ is equal to the eigenvalues of the matrix $\boldsymbol{\Psi}$. From definition of matrix $\boldsymbol{\Phi}$, there is one-to-one relationship between the parameters $\{\theta_n\}$.

We can simply estimate the matrix $\boldsymbol{\Psi}$ using least-squares (LS) approach:

$$\hat{\boldsymbol{\Psi}} = \underset{\boldsymbol{\Psi}}{\text{argmin}} \|\boldsymbol{U_2} - \boldsymbol{U_1}\boldsymbol{\Psi}\|_F = \boldsymbol{U_1}^{\dagger}\boldsymbol{U_2} \tag{2.34}$$

Then, we obtain the parameter estimates using the eigenvalues of the estimated $\hat{\boldsymbol{\Psi}}$ matrix.

Similar to MUSIC, ESPRIT calculates the eigenvalue decomposition of the sample correlation matrix, which has the computational complexity of $\mathcal{O}(T^3)$. Also, the estimation of the sample correlation matrix has the computational complexity of $\mathcal{O}(T^2L)$. The overall computational complexity of the ESPRIT method is $\mathcal{O}(T^3 + T^2L)$.

The main advantage of the ESPRIT method among the subspace-based methods is that there is no search procedure that inherent in the other subspace-based methods, including MUSIC. ESPRIT generates the parameter estimates directly from the eigenvalue decomposition of the sample correlation matrix.

CHAPTER 3

SPARSITY-BASED METHODS FOR PARAMETRIC RECONSTRUCTION

3.1 Introduction

Sparsity-based methods are compressed sensing techniques that are applied to sparse signal representation of parametric reconstruction problems. The basis of the compressed sensing theory is that if there is a prior knowledge about the sparsity of the unknown signal, the signal can be reconstructed using fewer samples than the sampling theorem requires [26]. Compressed sensing techniques solve the sparse approximation problems. In a sparse approximation problem, there is a target signal which can be represented by a linear weighted combination of a few elementary signals drawn from a signal dictionary [27]. Compressed sensing techniques approximates the target signal by finding these elementary basis signals. The general formulation of sparse approximation problem is

$$\text{find sparse } \boldsymbol{x} \quad \text{s.t.} \quad \boldsymbol{y} \approx \boldsymbol{A}\boldsymbol{x} \tag{3.1}$$

where $\boldsymbol{y}$ is the known target signal and $\boldsymbol{A}$ is the known dictionary matrix. This problem can be formulated as a combinatorial problem.

$$\min_{\boldsymbol{x}} \quad \|\boldsymbol{x}\|_0 \quad \text{s.t.} \quad \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}\|_2 \leq \epsilon \tag{3.2}$$

where ϵ is the error tolerance.

There are significant classes of compressed sensing techniques that solve Equation (3.1) computationally. [27] investigates these techniques in five major classes: greedy pursuit, convex relaxation, Bayesian framework, nonconvex optimization, and brute force. Among these classes, greedy pursuit and convex relaxation techniques have been extensively studied in the literature.

Greedy pursuit methods are first introduced in [28]. These methods approximate the target signal by identifying an elementary signal which is most related to the target signal from the dictionary at each iteration. Most well-known greedy pursuit approaches are matching pursuit (MP) [28], orthogonal matching pursuit (OMP) [29], and compressive sampling matching pursuit (CoSaMP) [30].

Equation (3.2) is a nonconvex problem due to ℓ_0-norm. The convex relaxation methods replaces the combinatorial problem in Equation (3.2) with a convex optimization problem. Some of the different convex relaxation methods are basis pursuit (BP) [31], focal underdetermined system solver (FOCUSS) [32], iterative re-weighted algorithm [33] and interior point methods [34], [27].

We investigate the sparsity-based methods in three main categories. These are on-grid, off-grid, and gridless sparse methods. The grid concept comes from the sparsity-based forward problem formulation of the parametric reconstruction, which will be discussed in the next section.

3.2 Sparsity-based Forward Problem

Parametric reconstruction problems with the classical forward model are mostly nonlinear parameter estimation problems because there is a nonlinear dependence between the parameters and measurements. The sparsity-based forward problem of parametric reconstruction is suggested in the literature to take advantage of the compressive sensing methods that solve the sparsity-based problems and linearize the nonlinear parameter estimation problem. Parametric reconstruction problems can be cast as sparse approximation problems using an overcomplete dictionary, which is obtained by sampling the continuous-valued parameter space. Sparsity-based forward problem formulation can be presented for both SMV and MMV cases.

3.2.1 SMV Case

Recall that the vector of a parametric function for n$^{\text{th}}$ source;

$$\boldsymbol{a}(\boldsymbol{\theta}_n) = [\phi(1, \boldsymbol{\theta}_n), ..., \phi(i, \boldsymbol{\theta}_n), ..., \phi(T, \boldsymbol{\theta}_n)]^T \tag{3.3}$$

where $\boldsymbol{\theta}_n = [\theta_{n_1} \ \theta_{n_2} \ ... \ \theta_{n_P}]^T$ and P is the number of parameters that belong to the parametric function. We obtain the candidate vectors of parametric functions by sampling the continuous-valued parameter space. We define a finite element parameter sequence in a P-dimensional space, $\{\bar{\boldsymbol{\theta}}_k\}_{k=1}^{K}$. The parameter vector using this sequence is $\bar{\boldsymbol{\theta}}_k = [\bar{\theta}_{k_1} \ \bar{\theta}_{k_2} \ ... \ \bar{\theta}_{k_P}]^T$ where $k_1 = 1, 2, ...K_1$, $k_2 = 1, 2, ...K_2$, ..., $k_P = 1, 2, ...K_P$ and $K = K_1 \times K_2 \times ... \times K_P$ is the total number of the elements of the sampling grid. The dictionary matrix is defined as follows:

$$\boldsymbol{A}(\bar{\boldsymbol{\Theta}}) = [a(\bar{\boldsymbol{\theta}}_1), ..., a(\bar{\boldsymbol{\theta}}_k), ..., a(\bar{\boldsymbol{\theta}}_K)] \in \mathbb{C}^{T \times K} \tag{3.4}$$

where each column of the matrix $\boldsymbol{A}(\bar{\boldsymbol{\Theta}})$ is an atom of the dictionary and represents the parametric function having the sampled parameter values. Also, the coefficient vector is defined as follows:

$$\boldsymbol{x} = [x_1, ..., x_k, ..., x_K]^T \in \mathbb{C}^K \tag{3.5}$$

Each entry of the coefficient vector belongs to the corresponding atom of the dictionary matrix. If the number of sampled points of the parameter space is large enough, i.e. $K \gg N$, then most of the entries of the coefficient vector will be zero. When all parameter values are in the set of sampled parameter space, then we can express the nonlinear parametric model using the dictionary matrix and the sparse vector $\boldsymbol{x}$. That is,

$$\boldsymbol{y} = \sum_{n=1}^{N} s_n \boldsymbol{a}(\boldsymbol{\theta_n}) + \boldsymbol{w} = \boldsymbol{A}(\bar{\boldsymbol{\Theta}})\boldsymbol{x} + \boldsymbol{w}, \tag{3.6}$$

where $\|\boldsymbol{x}\|_0 = N$. $\|\boldsymbol{x}\|_0$ is ℓ_0-norm of the vector $\boldsymbol{x}$ and it is equal to the number of nonzero entries in $\boldsymbol{x}$. The relation between the vector, $\boldsymbol{x}$ and the coefficients, $\{s_n\}$ is;

$$x_k = \begin{cases} s_n \ \text{ if } \quad \boldsymbol{\theta}_n = \bar{\boldsymbol{\theta}}_k \\ 0 \quad \text{otherwise} \end{cases} \tag{3.7}$$

We omit parameter sample sign $\bar{\boldsymbol{\Theta}}$ in Equation (3.6) for the sake of simplicity. Resulting sparse linear system representation is;

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{w} \tag{3.8}$$

where $\|\boldsymbol{x}\|_0 = N$.

The vector $\boldsymbol{x}$ is N-sparse but the nonzero entries do not appear arbitrarily. The sparse vector $\boldsymbol{x}$ has a block structure in terms of the placement of the nonzero entries. To see this, let us take $P = 2$ for simplicity and without loss of generality. In this case, the parameter vector consists of two parameters, $\boldsymbol{\theta}_n = [\theta_{n_1} \ \theta_{n_2}]^T$. We denote these parameters using different letters to avoid confusion, i.e. $[\theta_{n_1} \ \theta_{n_2}]^T = [\alpha \ \beta]$. The finite sequences $\{\bar{\alpha}_{k_1}\}_{k_1=1}^{K_1}$ and $\{\bar{\beta}_{k_2}\}_{k_2=1}^{K_2}$ consists of the sampled grid points of the parameters α and β. We obtain a dictionary matrix using the sampled grid points:

$$\boldsymbol{A}(\bar{\boldsymbol{\Theta}}) = [\boldsymbol{a}([\bar{\alpha}_1, \bar{\beta}_1]^T), ..., \boldsymbol{a}([\bar{\alpha}_{K_1}, \bar{\beta}_1]^T), ..., \boldsymbol{a}([\bar{\alpha}_1, \bar{\beta}_{K_2}]^T), ..., \boldsymbol{a}([\bar{\alpha}_{K_1}, \bar{\beta}_{K_2}]^T)] \tag{3.9}$$

The sparse vector that related to the above dictionary matrix is:

$$\boldsymbol{x} = [x_{1,1}, ..., x_{K_1,1}, x_{1,2}, ..., x_{K_1,2}, ..., x_{1,K_2}, ..., x_{K_1,K_2}]^T \tag{3.10}$$

We define the sub-vectors of the the sparse vector $\boldsymbol{x}$ using $\boldsymbol{x}^{k_2} = [x_{1,k_2}, ..., x_{K_1,k_2}]^T$ for $k_2 = 1, ..., K_2$. When k_2 is equal to the true value of the parameter β, the sub-vector $\boldsymbol{x}^{k_2}$ is a 1-sparse vector, i.e. $\|\boldsymbol{x}^{k_2}\|_0 \leq 1$, $k_2 = 1, 2, ..., K_2$. For the other k_2 values, the sub-vector is zero vector.

The sparse vector $\boldsymbol{x}$ is called a *block N-sparse* vector. *Block sparsity* means that the nonzero elements of the group or block sparse vector appear in blocks rather than being arbitrarily spread in compressive sensing literature [35]. In other words, a vector $\boldsymbol{x} \in \mathbb{C}^{K_1 K_2}$ is called block N-sparse vector if $\boldsymbol{x}^{k_2}$ has nonzero elements for at most N indices k_2 [36]. Note that, in case of $P = 1$, the discretization is performed over one parameter. Consequently, the size of the blocks in the sparse vector $\boldsymbol{x}$ will be one and the block sparsity will reduce to standard sparsity.

A number of properties on the dictionary matrix $\boldsymbol{A}$ in Equation (3.8) provides guarantee for efficient sparse recovery of the sparse vector $\boldsymbol{x}$, consequently the parameter vectors $\{\boldsymbol{\theta}_k\}_{k=1}^{K}$. Some of these properties are mutual coherence, restricted isometry property (RIP) [37] and null space property (NSP)[38].

Mutual coherence of the dictionary matrix $\boldsymbol{A}$ is defined as

$$\mu(\boldsymbol{A}) = \max_{i \neq j} \frac{|\boldsymbol{a}_i^H \boldsymbol{a}_j|}{\|\boldsymbol{a}_i\|_2 \|\boldsymbol{a}_j\|_2} \tag{3.11}$$

where $\boldsymbol{a}_i$ and $\boldsymbol{a}_j$ is the i^{th} and j^{th} column of the matrix $\boldsymbol{A}$, respectively [39]. The mutual coherence characterizes the dependence between columns of the dictionary

matrix $\boldsymbol{A}$ and it is used to guarantee stable sparse recovery. The smaller value of the mutual coherence provides efficient sparse recovery. However, smaller mutual coherence means that the number of samples on the discretization grid is small in our sparsity-based problem formulation. Hence, the possibility of the approximation errors due to grid mismatches increases with smaller mutual coherence.

As a result, the parametric models can be represented as sparse linear systems by sampling the continuous-valued parameter spaces. However, this discretization causes approximation errors when some of the true parameters do not lie in the sets of parameter samples. The approximation errors due to the grid mismatches in compressed sensing are investigated in [40]. Some of the methods for parametric reconstruction problems in the context of this thesis try to resolve the grid mismatch problem.

3.2.2 MMV Case

Recall that the dictionary matrix for SMV case is

$$\boldsymbol{A}(\bar{\boldsymbol{\Theta}}) = [a(\bar{\boldsymbol{\theta}}_1), ..., a(\bar{\boldsymbol{\theta}}_k), ..., a(\bar{\boldsymbol{\theta}}_K)] \in \mathbb{C}^{T \times K} \tag{3.12}$$

In the classical forward model, we have expressed both the general and the special MMV cases. In the general MMV case, the values of parameters change over different measurements. The classical forward model for this case is:

$$\boldsymbol{y}_l = \boldsymbol{A}(\boldsymbol{\Theta}_l)\boldsymbol{s}_l + \boldsymbol{n}_l \ , \ l = 1, 2, ..., L \tag{3.13}$$

In the special case, it is assumed that the parameter values are constant for all measurement vectors. Then, the classical forward model becomes:

$$\boldsymbol{Y} = \boldsymbol{A}(\boldsymbol{\Theta})\boldsymbol{S} + \boldsymbol{N} \tag{3.14}$$

The dictionary matrix in Equation (3.12) consists of all the columns of the matrix $\boldsymbol{A}(\boldsymbol{\Theta}_l)$ in Equation (3.13) as its atoms. As a result, we can use the dictionary matrix common for all measurement vectors. Then, the sparsity-based forward model is:

$$\boldsymbol{Y} = \boldsymbol{A}(\bar{\boldsymbol{\Theta}})\boldsymbol{X} + \boldsymbol{W} \tag{3.15}$$

where $\boldsymbol{Y} = [\boldsymbol{y}_1, ..., \boldsymbol{y}_L]$, $\boldsymbol{A}(\bar{\boldsymbol{\Theta}})$ is the dictionary matrix obtained as in SMV case, $\boldsymbol{X} = [\boldsymbol{x}_1, ..., \boldsymbol{x}_L]$ with $\|\boldsymbol{x}_l\|_0 = N$ for $l = 1, ..., L$, and $\boldsymbol{W} = [\boldsymbol{w}_1, ..., \boldsymbol{w}_L]$. The

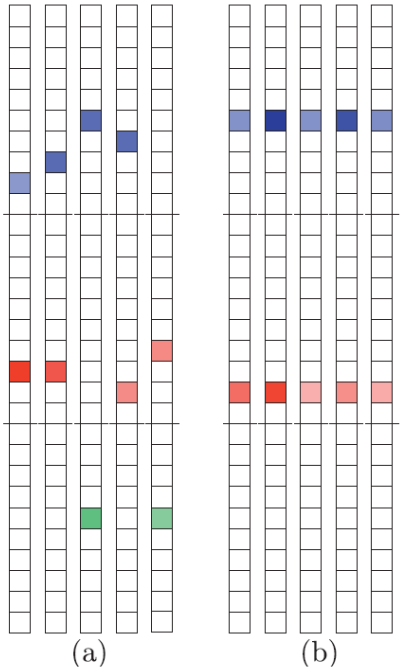

Figure 3.1: Demonstration of sparse matrix $\boldsymbol{X}$ with $N = 2$, $L = 5$, $K_1 = 10$ and $K_2 = 3$ where white squares represent the zero coefficients and colored squares represent nonzero coefficients (a) When there is a change of value of parameters with respect to index l (b) When there is no change of value of parameters with respect to index l which shows the joint sparsity

vectors $\boldsymbol{y}_l$, $\boldsymbol{x}_l$ and $\boldsymbol{w}_l$ have exactly the same form as in SMV case. We omit the parameter sample sign from Equation (3.15) for the sake of simplicity. The resultant sparsity-based MMV model is:

$$\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{X} + \boldsymbol{W} \tag{3.16}$$

Each sparse column vector of matrix $\boldsymbol{X}$ has the block sparse structure due to the discretization. Now, we discuss the relation between the sparse vectors in MMV case. If the parameter values are constant over different measurements, in other words, the special case in the classical forward model occurs, then each sparse vector shares a common support where the set of indices corresponding to the nonzero components is known as the support of the sparse vector. This concept is the special case of the block sparsity and called *joint sparsity* [41].

The alternative structures of the matrix $\boldsymbol{X}$ is given in Figure 3.1 when $P = 2$ for simplicity. Figure 3.1(a) shows that each column of the sparse matrix $\boldsymbol{X}$ is block N-sparse and has different supports. Figure 3.1(b) shows the jointly sparse column vectors. Note that each column vector is also block 2-sparse in Figure 3.1(b).

MMV sparse linear system representation also leads to approximation errors due to the discretization scheme. However, the use of sparse representation provides the utilization of many compressive sensing (CS) algorithms in the parametric reconstruction problems. Also, the CS community has started to develop algorithms that deal with the approximation error due to the discretization procedure. Some of the available algorithms will be presented in the following sections.

3.3 Inverse Problem

In the inverse problem, the goal is to recover all of the unknown parameter values, $\boldsymbol{\Theta}$, from the noisy measurements. The linear inverse problem is ill-posed because the discretization scheme causes an underdetermined system in the forward problem.

The linear least-squares approach is not reasonable for underdetermined linear systems [34]. The regularized least-squares approaches are appropriate for our problem since there exists a prior knowledge about the unknown solution, which is the sparsity. The general formulation of the regularized least-squares approach when solving the linear inverse problem in single measurement case is

$$\min_{\boldsymbol{x}} \quad \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}\|_2^2 + \lambda R(\boldsymbol{x}) \tag{3.17}$$

where the first term controls the data fidelity, $R(\boldsymbol{x})$ is a regularizer function, and λ is the regularization parameter that balances the data fidelity and regularizer terms. The solution of the inverse problem gives us the sparse vector $\boldsymbol{x}$. Estimated parameter values are in the atoms of the dictionary matrix corresponding to the indices of the nonzero entries of the sparse vector. As a result, we can estimate the parameter values by finding the sparse vector $\boldsymbol{x}$.

Similar to the above minimization problem, the constrained approach to the inverse

problem is:

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & R(\boldsymbol{x}) \\ \text{s.t.} \quad & \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}\|_2^2 \leq \epsilon \end{aligned} \tag{3.18}$$

where ϵ is the parameter that depends on the noise variance. Note that the problems (3.17) and (3.18) are equivalent for some values of λ and ϵ [42]. It is easy to set ϵ because it is proportional to the noise variance.

The extension of these problems to the MMV case is straightforward. For instance, the unconstrained regularized least-squares inverse problem in the MMV case is:

$$\min_{\boldsymbol{X}} \quad \|\boldsymbol{A}\boldsymbol{X} - \boldsymbol{Y}\|_2^2 + \lambda R(\boldsymbol{X}) \tag{3.19}$$

The selection of regularizer function, $R(.)$ depends on the prior knowledge about the unknown solution. The popular choices of the regularizer functions in case of sparsity prior are ℓ_0-norm, ℓ_1-norm in the SMV case and $\ell_{2,1}$-norm in the MMV case.

ℓ_0-norm counts the nonzero number of elements in the targeted vector in the SMV case. Minimizing ℓ_0-norm enforces the standard sparsity in the solution. The regularized least-squares with ℓ_0-norm is

$$\min_{\boldsymbol{x}} \quad \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}\|_2^2 + \lambda \|\boldsymbol{x}\|_0 \tag{3.20}$$

This inverse problem does not leverage the block structure of the sparse vector $\boldsymbol{x}$. Recall that the sparse vector has the block sparse structure when $P \geq 2$. If $\boldsymbol{x}$ is known as a block sparse vector, i.e., each block has the small number of nonzero entries, then this vector can be reconstructed by [35]

$$\min_{\boldsymbol{x}} \quad \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}\|_2^2 + \lambda \sum_{k_2=1}^{K_2} \mathcal{I}(\|\boldsymbol{x}^{k_2}\|_2) \tag{3.21}$$

where $\mathcal{I}(\|\boldsymbol{x}^{k_2}\|_2)$ is the indicator function that is 1 if $\|\boldsymbol{x}^{k_2}\|_2 > 0$ and $\boldsymbol{x}^{k_2} \in \mathbb{C}^{K_1}$ is the $k_2{}^{\text{th}}$ sub-block of $\boldsymbol{x}$. Note that we set $P = 2$ in Equation (3.21) with K_1 and K_2 being the number of discrete grid points for the two parameters.

However, the above optimization problems are not convex due to the ℓ_0-norms. Note that the indicator function in Equation (3.21) acts as an ℓ_0-norms because it counts the nonzero elements. The convex relaxation of the optimization problem in Equation

(3.20) is [27]

$$\min_{\boldsymbol{x}} \quad \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}\|_2^2 + \lambda\|\boldsymbol{x}\|_1 \tag{3.22}$$

and the convex relaxation of the inverse problem in Equation (3.21) is [35]

$$\min_{\boldsymbol{x}} \quad \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}\|_2^2 + \lambda \sum_{k_2=1}^{K_2} \|\boldsymbol{x}^{k_2}\|_2 \tag{3.23}$$

Remember that there exists joint sparsity in the unknown sparse matrix in the special case of MMV forward problem where the values of parameters remains unchanged over different measurements. The regularizer function that takes the joint sparsity into account is known as $\ell_{2,0}$-norm and its convex relaxation $\ell_{2,1}$-norm [16]. The resulting regularized least-squares problem using $\ell_{2,0}$-norm is

$$\min_{\boldsymbol{X}} \quad \|\boldsymbol{A}\boldsymbol{X} - \boldsymbol{Y}\|_F^2 + \lambda\|\boldsymbol{X}\|_{2,0} \tag{3.24}$$

where $\|\boldsymbol{X}\|_{2,0}$ counts the nonzero elements of the vector $[\|\boldsymbol{x}^1\|_2, ..., \|\boldsymbol{x}^l\|_2, ..., \|\boldsymbol{x}^L\|_2]^T$, $\boldsymbol{x}^l$ is the l$^{\text{th}}$ row vector of the sparse matrix $\boldsymbol{X}$ and $\|\cdot\|_F$ is called Frobenious-norm. Frobenious-norm is defined for a matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ as follows:

$$\|\boldsymbol{A}\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n} |a_{ij}|^2} \tag{3.25}$$

The convex relaxation of Equation (3.24) is

$$\min_{\boldsymbol{X}} \quad \|\boldsymbol{A}\boldsymbol{X} - \boldsymbol{Y}\|_F^2 + \lambda\|\boldsymbol{X}\|_{2,1} \tag{3.26}$$

where $\|\boldsymbol{X}\|_{2,1} = \sum_{l=1}^{L} \|\boldsymbol{x}^l\|_2$ and $\boldsymbol{x}^l$ is the l$^{\text{th}}$ row vector of the sparse matrix $\boldsymbol{X}$.

The dictionary matrix can increase the ill-posedness of the inverse problem in some cases. In Equation (3.8), the discretization is performed over all parameters in the parameter vector $\boldsymbol{\theta_n} = [\theta_{n_1}\ \theta_{n_2}\ ...\ \theta_{n_P}]^T$. However, in some cases, the discretization over some of the P parameters can lead to a highly coherent dictionary matrix $\boldsymbol{A}$. Thus, the ill-posedness of the sparse recovery increases. To solve this issue, we discretize some of the parameters, but the others are assumed to be known or estimated a priori because the latter parameters lead to a highly coherent dictionary.

Let us take $P = 2$ for simplicity and without loss of generality. The parameter vector consists of two parameters, $\boldsymbol{\theta_n} = [\alpha_n\ \beta_n]^T$. Also, assume that the discretization of

the parameter β leads to a highly coherent dictionary. As a result, the sparsity-based forward model is obtained by discretizing the parameter α_n and fixing the parameter β_n. Let $\{\bar{\alpha}_{k_1}\}_{k_1=1}^{K_1}$ be the set of possible parameter samples for the parameter α_n and fix the value of the parameter β_n for all n. The resulting dictionary matrix is given by

$$\begin{aligned}\tilde{\boldsymbol{A}}(\boldsymbol{\Theta}) = [a([\bar{\alpha}_1, \beta_1]^T), ..., a([\bar{\alpha}_{K_1}, \beta_1]^T), a([\bar{\alpha}_1, \beta_2]^T), ..., \\ a([\bar{\alpha}_{K_1}, \beta_2]^T), ..., a([\bar{\alpha}_1, \beta_N]^T), ..., a([\bar{\alpha}_{K_1}, \beta_N]^T)] \\ = [\tilde{\boldsymbol{A}}_1 ... \tilde{\boldsymbol{A}}_n ... \tilde{\boldsymbol{A}}_N]\end{aligned} \tag{3.27}$$

where β_1 to β_N are fixed variables, $\tilde{\boldsymbol{A}}_n = [a([\bar{\alpha}_1, \beta_n]^T), ..., a([\bar{\alpha}_{K_1}, \beta_n]^T)]$ for all k, and $\tilde{\boldsymbol{A}}(\boldsymbol{\Theta}) \in \mathbb{C}^{T \times K_1 N}$. The sparse vector related to this dictionary matrix in case of SMV is given by

$$\tilde{\boldsymbol{x}} = [x_{1,1}, ..., x_{K_1,1}, ..., x_{1,N}, ..., x_{K_1,N}]^T \tag{3.28}$$

where $\tilde{\boldsymbol{x}} \in \mathbb{C}^{K_1 N}$.

As a result, the sparsity-based SMV forward model is given by

$$\boldsymbol{y} = \tilde{\boldsymbol{A}}(\boldsymbol{\Theta})\tilde{\boldsymbol{x}} + \boldsymbol{w} \tag{3.29}$$

where $\|\tilde{\boldsymbol{x}}^n\|_0 = 1$ with $\tilde{\boldsymbol{x}}^n$ is the sub-vector of $\tilde{\boldsymbol{x}}$, i.e. $\tilde{\boldsymbol{x}}^n = [x_{1,n}, ..., x_{K_1,n}]^T$.

The inverse problem formulation related to Equation (3.29) is given by

$$\min_{\tilde{\boldsymbol{x}}, \boldsymbol{\beta}} \quad \|\tilde{\boldsymbol{A}}(\boldsymbol{\Theta})\tilde{\boldsymbol{x}} - \boldsymbol{y}\|_2^2 + \lambda R(\tilde{\boldsymbol{x}}) \tag{3.30}$$

where $\boldsymbol{\Theta} = [\bar{\boldsymbol{\alpha}}\ \boldsymbol{\beta}]^T$, $\boldsymbol{\alpha} = [\bar{\alpha}_1, ..., \bar{\alpha}_{k_1}, ..., \bar{\alpha}_{K_1}]^T$, $\boldsymbol{\beta} = [\beta_1, ..., \beta_n, ..., \beta_N]^T$ and $R(\tilde{\boldsymbol{x}})$ is a sparsity-inducing regularizer function. However, the inverse problem in Equation (3.30) is non-convex due to the parameter vector $\boldsymbol{\beta}$.

Alternating Least-Squares (ALS) approach provides an iterative two-stage alternating algorithm for the inverse problem in Equation (3.30). It separates the inverse problem into two sub-problems. The unknown parameter vector $\boldsymbol{\beta}$ is estimated in the first sub-problem using previously estimated sparse vector $\tilde{\boldsymbol{x}}$:

$$\min_{\boldsymbol{\beta}} \quad \|\tilde{\boldsymbol{A}}(\boldsymbol{\Theta})\tilde{\boldsymbol{x}} - \boldsymbol{y}\|_2^2 \tag{3.31}$$

Note that, β is treated as a continuous-valued parameter, and is not discretized in this case. The optimization problem in Equation (3.31) is similar to the optimization problem for nonlinear least-squares method which was discussed in Section 2. The optimization problem in Equation (3.31) can be solved via Levenberg-Marquardt or trust region algorithms.

Using the estimated parameter vector $\boldsymbol{\beta}$, the second sub-problem is written as follows:

$$\min_{\tilde{\boldsymbol{x}}} \quad \|\tilde{\boldsymbol{A}}(\boldsymbol{\Theta})\tilde{\boldsymbol{x}} - \boldsymbol{y}\|_2^2 + \lambda R(\tilde{\boldsymbol{x}}) \tag{3.32}$$

where $R(\tilde{\boldsymbol{x}})$ is a sparsity metric. The sparse recovery algorithms which will be presented in the next section can be used for the problem in Equation (3.32).

3.4 On-grid Sparse Methods

The first class of sparsity-based methods that solves the inverse problem is on-grid sparse methods. These methods directly apply compressed sensing techniques to parametric reconstruction problems. By doing so, the true parameters are assumed to lie on the set of parameter samples. First, we discuss the orthogonal matching pursuit and its modified version in greedy pursuit methods class because orthogonal matching pursuit is the most straightforward, effective greedy algorithm, and it is studied extensively in the compressive sensing literature [27]. Among the convex relaxation methods, basis pursuit denoising (BPDN), focal underdetermined system solver (FOCUSS), and ℓ_1-SVD will be discussed in this section.

3.4.1 SMV Case

3.4.1.1 Orthogonal Matching Pursuit (OMP)

OMP is a greedy pursuit method that iteratively refines the current estimate of the vector $\boldsymbol{x}$ by modifying few coefficients. Greedy methods heuristically solve an optimization problem by obtaining the local optimum solution at each iteration. However, there is no guarantee that the solution at the end of iterations is the global optimum solution. Greedy methods are a part of on-grid sparse methods since they estimate

only the components that are on the grid. Orthogonal matching pursuit is one of the most well-known and earliest on-grid methods for sparse approximation.

OMP numerically solves the inverse problem in Equation (3.20) which is

$$\min_{\boldsymbol{x}} \quad \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}\|_2^2 + \lambda\|\boldsymbol{x}\|_0 \tag{3.33}$$

Algorithm 1 shows the steps of OMP algorithm. Here, each column of $\boldsymbol{A}$ is denoted by $\boldsymbol{a}_k$ for $k = 1, 2, .., K$. $\boldsymbol{A}_\Omega$ denotes the sub-dictionary matrix where $\Omega \in \{1, 2, .., N\}$ and $\boldsymbol{A}_\Omega$ is updated for each iteration. ω represents the column index of the dictionary matrix $\boldsymbol{A}$.

Algorithm 1 Orthogonal Matching Pursuit

Require: $\boldsymbol{y}$: Measurements, $\boldsymbol{A}$: Dictionary Matrix,

Ensure: $\boldsymbol{x}$: Sparse Vector

1: $\mathbf{Initialization}$: Index set is $\Omega_0 = \emptyset$, the residual $\boldsymbol{r}_0 = \boldsymbol{y}$ and the counter is $t = 1$.

2: **repeat**

3: Find the index ω_t of the column $\boldsymbol{a}_k$ of $\boldsymbol{A}$ that is most correlated with the residual. Set

$$\omega_t = \underset{\boldsymbol{k}}{\text{argmax}} \quad | < \boldsymbol{r}_{t-1}, \boldsymbol{a}_k > | \quad \text{and} \quad \boldsymbol{\Omega}_t = \boldsymbol{\Omega}_{t-1} \cup \omega_t$$

4: Find the best coefficients for approximating the signal

$$\boldsymbol{x}_t = \underset{\boldsymbol{x}}{\text{argmin}} \quad ||\boldsymbol{y} - \boldsymbol{A}_{\Omega_t}\boldsymbol{x}||_2$$

5: Update the residual:

$$\boldsymbol{r}_t = \boldsymbol{y} - \boldsymbol{A}_{\Omega_t}\boldsymbol{x}_t$$

6: **until** Some convergence criterion is satisfied

OMP chooses the dictionary atom that is most correlated with the residual in each iteration, but it does not guarantee to obtain the optimum point at the end of the iterations. All of the greedy approaches have this drawback.

We will discuss a variant of the OMP algorithm that considers the block structure in the unknown sparse vector in the following section.

3.4.1.2 Block Orthogonal Matching Pursuit

Block OMP algorithm attempts to solve the sparse recovery problem having structured sparsity. It is the variant of the standard OMP algorithm that considers the sub-blocks of the sparse vector. The algorithm is applicable when the forward problem structure is as in Equation (3.29) which is

$$\boldsymbol{y} = \tilde{\boldsymbol{A}}(\boldsymbol{\Theta})\tilde{\boldsymbol{x}} + \boldsymbol{w} \tag{3.34}$$

It is known that each sub-block of the vector $\tilde{\boldsymbol{x}}$ has exactly one nonzero entry in Equation (3.29). Greedy algorithms are very effective when the number of nonzero entries in the sparse vector is known and small.

The constrained inverse problem related to the forward problem in Equation (3.29):

$$\begin{aligned} \min_{\tilde{\boldsymbol{x}}} \quad & \|\tilde{\boldsymbol{A}}\tilde{\boldsymbol{x}} - \boldsymbol{y}\|_2^2 \\ \text{s.t.} \quad & \|\tilde{\boldsymbol{x}}^n\|_0 \leq 1 \quad \text{for} \quad n = 1, 2.., N \end{aligned} \tag{3.35}$$

where $\tilde{\boldsymbol{x}}^n$ is the n^{th} sub-block of $\boldsymbol{x}$ that corresponds to $\tilde{\boldsymbol{A}}_n$ in Equation (3.27).

Block OMP algorithm is a greedy method that solves the inverse problem in Equation (3.35). The main idea of the block OMP algorithm is as follows:

- The atom of the dictionary $\boldsymbol{A}$ that is most correlated with the residual $\boldsymbol{r}_t$ is selected. If this atom is in $\tilde{\boldsymbol{A}}_n$, the index n is chosen. The selected source n is added to the selected sources list of Ψ so that the dictionary columns embedded in the blocks of $\boldsymbol{A}_n$ for sources $n \in \Psi$ will not be tested in next iterations. In this way, we can choose a single column from each block; the rest of the columns of the block will be ignored.

Algorithm 2 shows the steps of the block OMP algorithm, which deals with the block 1-sparse vectors.

3.4.1.3 Basis Pursuit Denoising (BPDN)

Basis pursuit denoising is a well-known sparse approximation approach that is applied to the linear sparse system representations of SMV models. It was first proposed in

Algorithm 2 Block Orthogonal Matching Pursuit

Require: $\boldsymbol{y}$: Measurements, $\tilde{\boldsymbol{A}} = [\tilde{\boldsymbol{A}}_1 ... \tilde{\boldsymbol{A}}_n ... \tilde{\boldsymbol{A}}_N]$: Dictionary Matrix,

Ensure: $\boldsymbol{x}$: Sparse Vector

1: **Initialization** : Index set is $\Omega_0 = \emptyset$, the residual $\boldsymbol{r}_0 = \boldsymbol{y}$ and the counter is $t = 1$.

2: **repeat**

3: Find the index ω_t of the column $\boldsymbol{a}_k$ of $\boldsymbol{A}$ that is most correlated with the residual. Set

$$\omega_t = \underset{\boldsymbol{k}}{\text{argmax}} \quad | < \boldsymbol{r}_{t-1}, \boldsymbol{a}_k > | \quad \text{and} \quad \boldsymbol{\Omega}_t = \boldsymbol{\Omega}_{t-1} \cup \omega_t$$

4: Choose the index of block $\boldsymbol{A}_n$ which includes $\boldsymbol{a}_{\omega_t}$. Add $n \in \Psi$. Then, use $\tilde{\boldsymbol{A}}$ by excluding the block $\boldsymbol{A}_n$. (Do not use columns of $\boldsymbol{A}_n$ in next iterations.)

5: Find the best coefficients for approximating the signal

$$\boldsymbol{x}_t = \underset{\boldsymbol{z}}{\text{argmin}} \quad ||\boldsymbol{y} - \boldsymbol{A}_{\Omega_t} \boldsymbol{z}||_2$$

6: Update the residual:

$$\boldsymbol{r}_t = \boldsymbol{y} - \boldsymbol{A}_{\Omega_n} \boldsymbol{x}_t$$

7: **until** Some convergence criterion is satisfied

[31]. Basis pursuit denoising method estimates the unknown sparse vector $\boldsymbol{x}$ from known noisy observation vector $\boldsymbol{y}$ and the dictionary matrix $\boldsymbol{A}$. The method belongs to the class of regularized least-squares methods. BPDN attempts to solve the inverse problem in Equation (3.22) which is

$$\min_{\boldsymbol{x}} \quad \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}\|_2^2 + \lambda \|\boldsymbol{x}\|_1 \tag{3.36}$$

This inverse problem is also called as least absolute shrinkage and selection operator (LASSO) in statistics community.

The numerical algorithms for BPDN approach use interior point [34], iterative shrinkage/thresholding [43] or alternating direction method of multipliers (ADMM) [44].

In this thesis, we use the alternating direction method of multipliers algorithm to solve the optimization problems in Equation (3.36) because ADMM has a number of useful properties as described below.

Alternating direction method of multipliers

ADMM is first introduced in [45]. It solves convex optimization problems by breaking the problem into smaller problems. ADMM has attracted considerable interest for the solution of convex optimization problems because it can be applied to non-differentiable objective functions [46]. Furthermore, it has a fast convergence rate in practice [47].

The general problem formulation is given by

$$\min_{\boldsymbol{x}} \quad f_1(\boldsymbol{x}) + f_2(\boldsymbol{G}\boldsymbol{x}) \tag{3.37}$$

.where $\boldsymbol{x} \in \mathbb{R}^k$ and $\boldsymbol{G} \in \mathbb{R}^{p \times k}$.

The following theorem explains the steps of the ADMM algorithm [45].

Theorem 1. *Consider f_1 and f_2 as convex, proper and closed functions and $\boldsymbol{G}$ have full column rank. Let $\mu \geq 0$ and $\boldsymbol{u}_0$, $\boldsymbol{d}_0 \in \mathbb{R}^p$ be arbitrary variables. Also, consider three sequences $\{\boldsymbol{x}_t\}$,$\{\boldsymbol{u}_t\}$ and $\{\boldsymbol{d}_t\}$ where t is the iteration number. If the following equations are satisfied, then $\{\boldsymbol{x}_t\}$ converges to the solution of Equation* (3.37).

$$\boldsymbol{x}_{t+1} = \min_{\boldsymbol{x}} \quad f_1(\boldsymbol{x}) + \frac{\mu}{2}\|\boldsymbol{G}\boldsymbol{x} - \boldsymbol{u}_t - \boldsymbol{d}_t\|_2^2 \tag{3.38}$$

$$\boldsymbol{u}_{t+1} = \min_{\boldsymbol{x}} \quad f_2(\boldsymbol{u}) + \frac{\mu}{2}\|\boldsymbol{G}\boldsymbol{x}_{t+1} - \boldsymbol{u}_t - \boldsymbol{d}_t\|_2^2 \tag{3.39}$$

$$\boldsymbol{d}_{t+1} = \boldsymbol{d}_t - (\boldsymbol{G}\boldsymbol{x}_{t+1} - \boldsymbol{u}_{t+1}) \tag{3.40}$$

Using the following translations, we can apply the ADMM algorithm to solve the inverse problem in Equation (3.22):

$$f_1(\boldsymbol{x}) = \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}\|_2^2 \tag{3.41}$$

$$f_2(\boldsymbol{x}) = \lambda\|\boldsymbol{x}\|_1 \tag{3.42}$$

$$\boldsymbol{G} \equiv \boldsymbol{I} \tag{3.43}$$

The minimization in Equation (3.38) is a strictly convex quadratic problem. The solution to this problem is unique and it is given by

$$\boldsymbol{x}_{t+1} = (\boldsymbol{A}^H \boldsymbol{A} + \mu \boldsymbol{I})^{-1}(\boldsymbol{A}^H \boldsymbol{y} + \mu(\boldsymbol{u}_t + \boldsymbol{d}_t)) \tag{3.44}$$

The solution of Equation (3.39) is given by the soft thresholding function:

$$\boldsymbol{u}_{t+1} = \mathrm{soft}(\boldsymbol{x}_{\mathrm{t}+1} - \boldsymbol{d}_{\mathrm{t}}, \lambda/\mu) \tag{3.45}$$

The soft-thresholding function, $\mathrm{soft}(\boldsymbol{v}, \tau) = \mathrm{sgn}(\boldsymbol{v}) \cdot \max(|\boldsymbol{v}| - \tau, 0)$ where $\mathrm{sgn}(.)$ equal to 1 for the positive numbers and it is equal to -1 for negative numbers. Note that the soft-thresholding function is calculated component-wise. Using the solution of the above minimization problems, the parameter $\boldsymbol{d}_t$ is updated using Equation (3.40). The resulting ADMM algorithm for the solution of Equation (3.22) is given in Algorithm 3.

Algorithm 3 ADMM for BPDN

Require: $\boldsymbol{y}$: Measurements, $\boldsymbol{A}$: Dictionary Matrix,

Ensure: $\boldsymbol{x}$: Sparse Vector

1: **Initialization** : Set the counter $t = 1$, choose $\boldsymbol{u}_0$, $\boldsymbol{d}_0$ and $\mu > 0$.

2: **repeat**

3:
$$\boldsymbol{x}_{t+1} = (\boldsymbol{A}^H \boldsymbol{A} + \mu \boldsymbol{I})^{-1}(\boldsymbol{A}^H \boldsymbol{y} + \mu(\boldsymbol{u}_t + \boldsymbol{d}_t))$$

4:
$$\boldsymbol{u}_{t+1} = \mathrm{soft}(\boldsymbol{x}_{\mathrm{t}+1} - \boldsymbol{d}_{\mathrm{t}}, \lambda/\mu) = \mathrm{sgn}(\boldsymbol{x}_{\mathrm{t}+1} - \boldsymbol{d}_{\mathrm{t}}) \cdot \max(|\boldsymbol{x}_{\mathrm{t}+1} - \boldsymbol{d}_{\mathrm{t}}| - \lambda/\mu)$$

5:
$$\boldsymbol{d}_{t+1} = \boldsymbol{d}_t - (\boldsymbol{x}_{t+1} - \boldsymbol{u}_{t+1})$$

6: **until** Some convergence criterion is satisfied

3.4.1.4 Focal Underdetermined System Solver (FOCUSS)

Focal Underdetermined System Solver (FOCUSS) is a re-weighted minimum norm algorithm that solves the SMV parametric reconstruction problems using sparse signal representation [32]. It starts by finding a non-sparse low-resolution estimate of

$\boldsymbol{x}$ in the sparse signal representation of the SMV parametric problem, which is given below. Then, it estimates the signal $\boldsymbol{x}$ sparsely in the next iterations.

Consider the sparse signal representation of SMV parametric model in Equation (3.8). FOCUSS solves the inverse problem in Equation (3.20), indirectly. The algorithm iteratively finds a weighted minimum norm solution to following objective function:

$$\begin{aligned} &\text{find } \boldsymbol{x} = \boldsymbol{W}\boldsymbol{q} \\ &\text{where } \boldsymbol{q} = \arg\min_{\boldsymbol{q}} ||\boldsymbol{q}||_2^2 \quad \text{s.t.} \quad ||\boldsymbol{y} - \boldsymbol{A}\boldsymbol{W}\boldsymbol{q}||_2^2 \leq \epsilon \end{aligned} \tag{3.46}$$

The above objective function can be written in unconstrained regularized least squares form:

$$\begin{aligned} &\text{find } \boldsymbol{x} = \boldsymbol{W}\boldsymbol{q} \\ &\text{where } \boldsymbol{q} = \arg\min_{\boldsymbol{q}} ||\boldsymbol{q}||_2^2 + \beta^2 ||\boldsymbol{y} - \boldsymbol{A}\boldsymbol{W}\boldsymbol{q}||_2^2 \end{aligned} \tag{3.47}$$

To solve the objective function in Equation (3.47) at iteration t , Tikhonov's regularization method [48] can be used:

$$\hat{\boldsymbol{x}}_t = \boldsymbol{W}_t \hat{\boldsymbol{q}}_t = \boldsymbol{W}_t (\boldsymbol{W}_t^H \boldsymbol{A}^H \boldsymbol{A} \boldsymbol{W}_t + \beta^{-2} \boldsymbol{I})^{-1} \boldsymbol{W}_t^H \boldsymbol{A}^H \boldsymbol{y} \tag{3.48}$$

Denote the estimate of the vector $\boldsymbol{x}$ at (t-1)$^{\text{th}}$ iteration $\hat{\boldsymbol{x}}_{\boldsymbol{t-1}} = [\hat{x}_{t-1}(1)...\hat{x}_{t-1}(K)]^T$. Then, the weight matrix at the current iteration t is defined as, $\boldsymbol{W}_t = [\text{diag}(\hat{\boldsymbol{x}}_{\boldsymbol{t-1}})^p]$ where the power p is considered as component-wise and $0.5 \leq p \leq 1$. As an initial condition, Tikhonov's regularization method can also be used:

$$\hat{\boldsymbol{x}}_1 = \min ||\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}||_2^2 + \alpha^2 ||\boldsymbol{x}||_2^2 = (\boldsymbol{A}^H \boldsymbol{A} + \alpha^2 \boldsymbol{I})^{-1} \boldsymbol{A}^H \boldsymbol{y} \tag{3.49}$$

Algorithm 4 shows the steps of the FOCUSS algorithm. The algorithm is an iterative method that is appropriate for finding solutions to underdetermined linear systems. It is well suited to parametric reconstruction problems because the sparse signal representation of parametric reconstruction problems can be seen as an instance of the underdetermined linear systems.

The main complexity arises from Tikhonov's regularization in Equation (3.48) and it amounts $\mathcal{O}(T^3)$. The total computational complexity of FOCUSS algorithm is $\mathcal{O}(T^3 + T^2 K)$ [32].

Algorithm 4 Focal Underdetermined System Solver

1: **Initialization** :

$$\hat{\boldsymbol{x}}_1 = \min ||\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}||_2^2 + \alpha^2 ||\boldsymbol{x}||_2^2 = (\boldsymbol{A}^H \boldsymbol{A} + \alpha^2 \boldsymbol{I})^{-1} \boldsymbol{A}^H \boldsymbol{y}$$

2: **repeat**

3: Form the weighting matrix using $\boldsymbol{W}_t = [\text{diag}(\hat{\boldsymbol{x}}_{\boldsymbol{t-1}})]$

4: Solve the following equation.

$$\hat{\boldsymbol{x}}_t = \boldsymbol{W}_t \hat{\boldsymbol{q}}_t = \boldsymbol{W}_t (\boldsymbol{A}\boldsymbol{W}_t)^+ \boldsymbol{y} = \boldsymbol{W}_t (\boldsymbol{W}_t^H \boldsymbol{A}^H \boldsymbol{A} \boldsymbol{W}_t + \beta^2 \boldsymbol{I})^{-1} \boldsymbol{W}_t^H \boldsymbol{A}^H \boldsymbol{y}$$

5: **until** Some convergence criterion is satisfied

3.4.1.5 Discussions

In this section, we review the on-grid sparse methods that are applied to SMV parametric recovery problems. OMP and Block OMP methods are in the class of greedy algorithms. BPDN and FOCUSS methods belong to the class of convex optimization algorithms. Although the greedy algorithms are significantly faster than BPDN and FOCUSS, they have weak performance guarantees, unlike the convex optimization methods.

Block OMP enforces additional prior information with respect to OMP, which is the block sparsity. As a result, block OMP possibly outperforms OMP because parametric recovery problems have block sparsity structure. Note that if there is one parameter in the problem, the block sparsity reduces to standard sparsity and both methods gives the same results.

FOCUSS attempts to solve the SMV parametric reconstruction problems using sparse signal representation. The sparsity is enforced implicitly in the objective function using ℓ_2-norm minimization. However, BPDN enforces sparsity explicitly in the objective function using ℓ_1-norm.

3.4.2 MMV Case

3.4.2.1 ℓ_1-SVD

ℓ_1-SVD algorithm attempts to solve the MMV parametric reconstruction problems using sparse signal representation. It can be seen as MMV extension of basis pursuit denoising method because ℓ_1-SVD attempts to solve regularized least-squares minimization problems as in the case of BPDN. Recall that the sparse signal representation for MMV parametric model when joint sparsity holds is given by

$$\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{X} + \boldsymbol{W} \tag{3.50}$$

The nonzero entries of each column of matrix $\boldsymbol{X}$ represent the coefficients of superimposed signals over a time sample. The method assumes that values of parameters of the superimposed parametric signals are stationary over time samples. The stationarity means that each column of matrix $\boldsymbol{X}$ has the same sparsity pattern, i.e., the columns are jointly sparse. The matrix $\boldsymbol{X}$ is parameterized both spatially and temporally, but sparsity is enforced only spatially. The matrix $\boldsymbol{X}$ has jointly sparse column vectors where each column of the matrix has the same support set. ℓ_1-SVD attempts to solve the inverse problem in Equation (3.26).

The main disadvantage of this technique is the computational cost when L is large. To solve this problem, ℓ_1-SVD algorithm suggests a dimensionality reduction technique. The algorithm uses singular value decomposition to decompose the measurement matrix into the signal and noise subspaces, then omits the noise subspace.

In a noiseless sparse signal representation, number of L measurement vectors, $\{\boldsymbol{y}(l)\}_{l=1}^{L}$ lie in a L-dimensional subspace. The algorithm takes a basis for L-dimensional subspace to estimate which combinations of columns of the matrix $\boldsymbol{A}$ form measurements. Let us take the singular value decomposition of the measurement matrix $\boldsymbol{Y}$:

$$\boldsymbol{Y} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^H \tag{3.51}$$

where $\boldsymbol{U}$ and $\boldsymbol{V}$ are orthogonal and $\boldsymbol{Y} \in \mathbb{C}^{T\times L}$, $\boldsymbol{U} \in \mathbb{C}^{T\times T}$, $\boldsymbol{\Sigma} \in \mathbb{C}^{T\times L}$, $\boldsymbol{V} \in \mathbb{C}^{L\times L}$. We define a matrix that combination of an identity matrix and a zero matrix:

$$\boldsymbol{D}_N = [\boldsymbol{I}_N \boldsymbol{0}_{\boldsymbol{N}\times(\boldsymbol{L}-\boldsymbol{N})}]^T \tag{3.52}$$

where $\boldsymbol{I}_N$ is $N \times N$ identity matrix, $\mathbf{0}_{\boldsymbol{N}\times(\boldsymbol{L}-\boldsymbol{N})}$ is a matrix of zeros and N is the number of nonzero rows in the sparse matrix. The problem size decreases from L to N using the dimensionality reduction via SVD.

The reduced dimensional matrices of the measurement matrix $\boldsymbol{Y}$, sparse coefficient matrix $\boldsymbol{X}$ and the noise matrix $\boldsymbol{W}$ are defined as follows:

$$\begin{aligned}\boldsymbol{Y}_{SV} &= \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{D}_N = \boldsymbol{Y}\boldsymbol{V}\boldsymbol{D}_N \\ \boldsymbol{X}_{SV} &= \boldsymbol{X}\boldsymbol{V}\boldsymbol{D}_N \\ \boldsymbol{W}_{SV} &= \boldsymbol{W}\boldsymbol{V}\boldsymbol{D}_N\end{aligned}$$

The reduced dimension counterpart of MMV parametric model is given by

$$\boldsymbol{Y}_{SV} = \boldsymbol{A}\boldsymbol{X}_{SV} + \boldsymbol{W}_{SV} \tag{3.53}$$

where $\boldsymbol{Y}_{SV} \in \mathbb{C}^{TxN}$, $\boldsymbol{X}_{SV} \in \mathbb{C}^{KxN}$ and $\boldsymbol{W}_{SV} \in \mathbb{C}^{TxN}$ The objective function that is related to the above model is given by

$$\hat{\boldsymbol{X}}_{SV} = \arg\min_{\boldsymbol{X}_{SV}} \quad ||\boldsymbol{Y}_{SV} - \boldsymbol{A}\boldsymbol{X}_{SV}||_F^2 + \lambda||\boldsymbol{X}_{SV}||_{2,1} \tag{3.54}$$

As a result, the L-dimensional parametric reconstruction problem is converted to N-dimensional problem by using singular value decomposition where $L \gg N$.

The second-order cone (SOC) programming optimization method is suggested for the solution of the inverse problem in Equation (3.54). The reason for this choice is that the SOC programming has a substantial theoretical foundation, and it has available efficient interior-point algorithms.

The objective function in Equation (3.54) can be written as a second order cone programming problem [49]. To do this, we put the non-linearities into the constraints and linearize the objective function first:

$$\begin{aligned}\min \quad & p + \lambda q \\ \text{s.t.} \quad & ||\boldsymbol{Y}_{SV} - \boldsymbol{A}\boldsymbol{X}_{SV}||_F^2 \leq\ p \quad \text{and} \quad ||\boldsymbol{X}_{SV}||_{2,1} \leq\ q\end{aligned} \tag{3.55}$$

The constraints in Equation (3.55) can be expressed in different way. Let's write the first constraint term as a quadratic constraint:

We define a vector $\boldsymbol{z}_k = \boldsymbol{y}_{SV,k} - \boldsymbol{A}\boldsymbol{x}_{SV,k}$ where $\boldsymbol{y}_{SV,k}$ and $\boldsymbol{x}_{SV,k}$ are the k$^{\text{th}}$ columns of $\boldsymbol{Y}_{SV}$ and $\boldsymbol{X}_{SV}$, respectively. Also, we define a new vector by stacking the vectors

$\boldsymbol{z}_k$: $\boldsymbol{z} = \begin{bmatrix} \boldsymbol{z}_1 \\ : \\ \boldsymbol{z}_K \end{bmatrix}$. Then, $||\boldsymbol{z}||_2^2 \leq p$. The second constraint term can be written in linear summation of quadratic terms. Let $||\boldsymbol{X}_{SV}||_{2,1} = \mathbf{1}^H \boldsymbol{x}_{SV}^{(l_2)}$ where $\boldsymbol{x}_{SV}^{(l_2)}$ is a vector that each entry is the l_2-norm of the respective row of the matrix $\boldsymbol{X}_{SV}$. $\mathbf{1}^H \boldsymbol{x}_{SV}^{(l_2)} \leq q$ inequality can be re-written in the following form:

$$\sqrt{\sum_{k=1}^{K} (x_{SV,k}(i)^2)} \leq r_i \quad \text{and} \quad \mathbf{1}^H \boldsymbol{r} \leq q \quad \text{where} \quad \boldsymbol{r} = [r_1 ..., r_i, ..., r_T]^H \tag{3.56}$$

As a result, the SOC programming form of the optimization problem in Equation (3.54) is given by

$$\begin{aligned} \min \quad & p + \lambda q \\ \text{s.t.} \quad & ||\boldsymbol{z}||_2^2 \leq p \quad \text{and} \quad \mathbf{1}^H \boldsymbol{r} \leq q \end{aligned} \tag{3.57}$$

The minimization problem in Equation (3.57) can be numerically optimized with interior point algorithms in [50].

$\boldsymbol{\ell_1}$-SVD is a solution method to parametric reconstruction problems that consider the parametric signal model in a sparse signal representation framework. The method uses singular value decomposition to decrease the computational cost when L is large in the MMV parametric problem. The related inverse problem is optimized using second-order cone programming. The main advantages of this method over the previous ones are parameter spacing resolution increase, robustness to noise, and robustness to the correlation of the superimposed signals.

The computational complexity of the $\boldsymbol{\ell_1}$-SVD is $\mathcal{O}(K^3 N^3)$, and it is independent of the number of the measurement L [51]. Compared to the classical methods, $\boldsymbol{\ell_1}$-SVD has higher computational complexity because it solves a joint optimization problem.

3.5 Off-Grid Sparse Methods

Off-grid sparse methods are proposed in the literature to resolve the grid mismatch problem encountered in on-grid sparse methods. In these methods, a grid is still necessary to apply the sparsity-based algorithms. However, these methods consider the parameter samples out of the grid points by changing the grid points at each iteration

or embedding them into the sparse signal model. Off-grid sparse methods can be divided into two classes. The first class uses the off-grid sparse signal representation, which will be explained in the following section. The second class relies on the dynamic dictionary or grid, i.e., jointly updates the grid points and the sparse vector. We have selected two representative methods from each class, off-grid sparse Bayesian learning, and a super-resolution iterative re-weighted method. Both methods attempt to solve the inverse problem in Equation (3.22).

3.5.1 SMV Case

3.5.1.1 Super-resolution Iterative Re-weighted Method

The concept of super-resolution is introduced in [52] as the ability of resolving the true parameters with infinite precision. As an off-grid sparse method, super-resolution iterative re-weighted (SURE-IR) method considers all parameter space unlike the on-grid sparse methods [13]. The method suggests to iteratively update the grid points in sparse signal representation to minimize the approximation errors. Moreover, this method is applicable only to SMV parametric reconstruction problems having one parameter in the parametric function. The sparse signal representations for SMV model is written as follows:

$$\boldsymbol{y} = \boldsymbol{A}(\bar{\boldsymbol{\Theta}})\boldsymbol{x} + \boldsymbol{n} \tag{3.58}$$

where $||\boldsymbol{x}||_0 = K$. In the above forward problem, there are P parameters that identifies the parametric function. If we select $P = 1$, then the $\bar{\boldsymbol{\Theta}}$ matrix reduces to the row vector $\bar{\boldsymbol{\theta}} = [\theta_1, ..., \theta_k, ...\theta_K]$. The above sparse representation is formulated as an inverse problem using $\bar{\boldsymbol{\theta}}$:

$$\begin{aligned} \min_{\boldsymbol{x},\bar{\boldsymbol{\theta}}} \quad & ||\boldsymbol{x}||_0 \\ \text{s.t.} \quad & ||\boldsymbol{y} - \boldsymbol{A}(\bar{\boldsymbol{\theta}})\boldsymbol{x}||_2 \leq \epsilon \end{aligned} \tag{3.59}$$

where the matrix $\boldsymbol{A}(\bar{\boldsymbol{\theta}})$ is iteratively updated unlike the on-grid sparse methods. However, the optimization in Equation (3.59) is an NP-hard problem. Instead of ℓ_0-norm, most of compressed sensing methods suggest ℓ_1-norm to make the optimization computationally efficient including BPDN and ℓ_1-SVD. This method suggest an alter-

native sparsity promoting log-sum penalty function. By replacing the ℓ_0-norm in Equation (3.59) with log-sum penalty function, the new optimization becomes

$$\begin{aligned} \min_{\boldsymbol{x},\bar{\boldsymbol{\theta}}} \quad & L(\boldsymbol{x}) = \sum_{k=1}^{K} \log(|x_k|^2 + \tau) \\ \text{s.t.} \quad & ||\boldsymbol{y} - \boldsymbol{A}(\bar{\boldsymbol{\theta}})\boldsymbol{x}||_2 \leq \epsilon \end{aligned} \tag{3.60}$$

where x_k is the k$^{\text{th}}$ entry of sparse vector $\boldsymbol{x}$ and τ is a positive parameter to guarantee that the log-sum function is well-defined.

The unconstrained form of the above optimization problem is given by

$$\min_{\boldsymbol{x},\bar{\boldsymbol{\theta}}} \quad \sum_{k=1}^{K} \log(|x_k|^2 + \tau) + \lambda||\boldsymbol{y} - \boldsymbol{A}(\bar{\boldsymbol{\theta}})\boldsymbol{x}||_2^2 \tag{3.61}$$

This optimization problem can be solved via an algorithm that iteratively alternates between an update of the sparse vector and the parameters. However, these methods do not guarantee that objective function value monotonically decreases at each iteration. The iterative re-weighted method proposes a majorization-minimization approach [33] to solve Equation (3.61). Instead of log-sum function in Equation (3.61), use a convex and differentiable surrogate function majorizing the log-sum function. The proposed surrogate function is given by

$$Q(\boldsymbol{x}|\hat{\boldsymbol{x}}_t) = \sum_{k=1}^{K} \left(\frac{|x_k|^2 + \tau}{|(\hat{x}_k)_t|^2 + \tau} + \log(|(\hat{x}_k)_t|^2 + \tau) - 1 \right) \tag{3.62}$$

where subscript t denotes the current iteration number. Then, the optimization problem in Equation (3.61) becomes

$$S(\boldsymbol{x}, \bar{\boldsymbol{\theta}}|\hat{\boldsymbol{x}}_t) = Q(\boldsymbol{x}|\hat{\boldsymbol{x}}_t) + \lambda||\boldsymbol{y} - \boldsymbol{A}(\bar{\boldsymbol{\theta}})\boldsymbol{x}||_2^2 \tag{3.63}$$

If we ignore terms independent of $\boldsymbol{x}$ and $\bar{\boldsymbol{\theta}}$, the above objective function can be simplified as follows:

$$\min_{\boldsymbol{x},\bar{\boldsymbol{\theta}}} \quad \boldsymbol{x}^H \boldsymbol{D}_t \boldsymbol{x} + \lambda||\boldsymbol{y} - \boldsymbol{A}(\bar{\boldsymbol{\theta}})\boldsymbol{x}||_2^2 \tag{3.64}$$

where $\boldsymbol{D}_t = \text{diag}[\frac{1}{|(\hat{x}_1)_t|^2+\tau}, \frac{1}{|(\hat{x}_2)_t|^2+\tau}, ..., \frac{1}{|(\hat{x}_K)_t|^2+\tau}]$. The optimal $\boldsymbol{x}$ is the least squares solution of the problem in Equation (3.64) when fixing $\bar{\boldsymbol{\theta}}$:

$$(\boldsymbol{A}^H(\bar{\boldsymbol{\theta}})\boldsymbol{A}(\bar{\boldsymbol{\theta}}) + \lambda^{-1}\boldsymbol{D}_t)^{-1}\boldsymbol{A}^H(\bar{\boldsymbol{\theta}})\boldsymbol{y} \tag{3.65}$$

Substituting Equation (3.65) into Equation (3.64), we get the objective function that will be minimized over $\bar{\boldsymbol{\theta}}$ while fixing $\boldsymbol{x}$. However, there is no analytical solution for this optimization problem. Since the objective function is differentiable, we use gradient descent method to obtain an estimate for $\bar{\boldsymbol{\theta}}$. As a result, we obtain a non-increasing objective function $S(\boldsymbol{x}_{t+1}, \bar{\boldsymbol{\theta}}_{t+1}|\hat{\boldsymbol{x}}_t)$. As a result of iterations that alternates between finding optimum solution to sparse vector $\boldsymbol{x}$ and decreasing the objective function using gradient descent method, $S(\boldsymbol{x}_{t+1}, \bar{\boldsymbol{\theta}}_{t+1}|\hat{\boldsymbol{x}}_t)$ eventually converges to a stationary point. The algorithm stops when the difference $\boldsymbol{x}_{t+1} - \boldsymbol{x}_t$ is smaller than a prescribed tolerance value.

3.5.2 MMV Case

3.5.2.1 Off-Grid Sparse Bayesian Inference (OGSBI)

Sparse signal representation of parametric reconstruction problems assumes that the set of parameter samples consists of all the true parameters. However, this is not guaranteed, and the grid selection will inevitably end up grid mismatch. Off-grid sparse signal representation modifies the dictionary matrix introducing grid offset parameters so that the parameter samples out of the grid points are allowed to be estimated.

Let the parameter vector consist of a single parameter, i.e. $\boldsymbol{\theta_n} = \theta_n$ for simplicity. The set of grid points is $\{\bar{\theta}_k\}_{k=1}^K$. Assume that the grid points are uniformly spaced. Then, the grid interval is defined as $r = \bar{\theta}_k - \bar{\theta}_{k-1}$. Suppose the nearest grid point to the actual parameter value θ_n is $\bar{\theta}_{k_n}$. We approximate the column vectors in the dictionary matrix using a first order Taylor's expansion:

$$\boldsymbol{a}(\theta_n) = \boldsymbol{a}(\bar{\theta}_{k_n}) + \boldsymbol{b}(\bar{\theta}_{k_n})(\theta_n - \bar{\theta}_{k_n}) \tag{3.66}$$

where $\boldsymbol{b}(\bar{\theta}_{k_n})$ is the derivative of $\boldsymbol{a}(\bar{\theta}_{k_n})$.

The grid offset is defined for each $\bar{\theta}_k$ as $\{\beta_k\}_{k=1}^K \in [-0.5r, 0.5r]$. Then the off-grid sparse representation for SMV model is given by

$$\boldsymbol{y} = \boldsymbol{\Phi}(\boldsymbol{\beta})\boldsymbol{x} + \boldsymbol{w} \tag{3.67}$$

where $\boldsymbol{\Phi}(\boldsymbol{\beta}) = \boldsymbol{A} + \boldsymbol{B}[\text{diag}(\boldsymbol{\beta})]$, $\boldsymbol{B} = [\boldsymbol{b}(\bar{\theta}_1) \ ... \ \boldsymbol{b}(\bar{\theta}_K)]$ with

$$\begin{aligned} x_k &= \begin{cases} s_n \ \text{ if } \bar{\theta}_k = \bar{\theta}_{k_n} \\ 0 \quad \text{otherwise} \end{cases} \\ \beta_k &= \begin{cases} \theta_n - \bar{\theta}_{k_n} \ \text{ if } \bar{\theta}_k = \bar{\theta}_{k_n} \\ 0 \quad \text{otherwise} \end{cases} \end{aligned} \tag{3.68}$$

The off-grid sparse representation for MMV model is obtained by stacking the measurement vectors, sparse coefficient vectors and noise vectors side by side:

$$\boldsymbol{Y} = \boldsymbol{\Phi}(\boldsymbol{\beta})\boldsymbol{X} + \boldsymbol{W} \tag{3.69}$$

Off-grid sparse representations introduce additional grid offset parameters to be estimated. The computational cost of the solution methods possibly increases with the benefit of more accurate estimation of parameter samples of superimposed signals. Note that Equation (3.67) and (3.69) reduces to standard sparse signal representation of SMV and MMV parametric problems if $\boldsymbol{\beta} = \boldsymbol{0}$, respectively.

In case of off-grid sparse representation of MMV parametric model, the inverse problem in Equation (3.26) converts into the following problem:

$$\min_{\boldsymbol{X},\boldsymbol{\beta}} \quad \|\boldsymbol{\Phi}(\boldsymbol{\beta})\boldsymbol{X} - \boldsymbol{Y}\|_2^2 + \lambda\|\boldsymbol{X}\|_{2,1} \tag{3.70}$$

Note that the vector $\boldsymbol{\beta}$ in $\boldsymbol{\Phi}(\boldsymbol{\beta})$ is also unknown in addition to $\boldsymbol{X}$ in the above inverse problem.

The OGSBI method uses sparse Bayesian learning approach to the off-grid sparse representations of MMV parametric reconstruction problems [53]. OGSBI attempts to solve the inverse problem in Equation (3.70) from Bayesian perspective. Assume that the off-grid sparse MMV model has circular symmetric complex Gaussian noise. Then the noise can be modeled as follows:

$$p(\boldsymbol{W}|\alpha_0) = \prod_{l=1}^{L} CN(\boldsymbol{w_l}|\boldsymbol{0}, \alpha_0^{-1}\text{I}) \tag{3.71}$$

where $\alpha_0 = \sigma^{-2}$ is the noise precision with σ^2 being the noise variance and the complex Gaussian distribution of the random vector $\boldsymbol{c}$ with the mean $\boldsymbol{\mu}$ and the covariance Σ is given by

$$CN(\boldsymbol{c}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\pi^K|\boldsymbol{\Sigma}|}\exp\{-(\boldsymbol{c}-\boldsymbol{\mu})^H\boldsymbol{\Sigma}^{-1}(\boldsymbol{c}-\boldsymbol{\mu})\} \tag{3.72}$$

Recall that the off-grid sparse representation of parametric reconstruction problems is given by

$$\boldsymbol{Y} = \boldsymbol{\Phi}(\boldsymbol{\beta})\boldsymbol{X} + \boldsymbol{W} \tag{3.73}$$

Using Equation (3.73) and (3.71), we have

$$p(\boldsymbol{Y}|\boldsymbol{X}, \alpha_0, \boldsymbol{\beta}) = \prod_{l=1}^{L} CN(\boldsymbol{y_l}|\boldsymbol{\Phi}(\boldsymbol{\beta})\boldsymbol{x_l}, \alpha_0^{-1}\mathrm{I}) \tag{3.74}$$

We assume that the noise precision α_0 is unknown and a Gamma distribution is assigned for noise precision, i.e. $p(\alpha_0; c, d) = \Gamma(\alpha_0; c, d)$ with c and d being the Gamma function parameters and the values of these parameters are selected close to zero because the prior is uninformative.

We know that the coefficient matrix $\boldsymbol{X}$ is joint sparse in Equation (3.73). In sparse Bayesian inference, sparsity is induced with a two-level hierarchical complex Gaussian prior [54]. Assume that the columns of the coefficient matrix $\boldsymbol{X}$ are independent and assign the hierarchical prior to this matrix:

$$p(\boldsymbol{X}; \rho) = \int p(\boldsymbol{X}|\boldsymbol{\alpha})p(\boldsymbol{\alpha}; \rho)\mathrm{d}\boldsymbol{\alpha} \tag{3.75}$$

where ρ is a non-negative number and $\boldsymbol{\alpha} \in \mathbb{R}^N$. The probability density functions in the integral in Equation (3.75) are given by

$$p(\boldsymbol{X}|\boldsymbol{\alpha}) = \prod_{l=1}^{L} CN(\boldsymbol{x_l}|\boldsymbol{0}, \boldsymbol{\Lambda}) \quad \text{and} \quad p(\boldsymbol{\alpha}; \rho) = \prod_{k=1}^{K} \Gamma(\alpha_k|1, \rho) \tag{3.76}$$

where $\boldsymbol{\Lambda} = \mathrm{diag}(\boldsymbol{\alpha})$. The only prior information about the grid offset vector $\boldsymbol{\beta}$ is its boundedness. Then, we can assume a uniform distribution for the grid offset vector:

$$\boldsymbol{\beta} \sim U\left(\left[-\frac{r}{2}, \frac{r}{2}\right]^K\right) \tag{3.77}$$

Combining the above PDFs of the Bayesian model, the joint density is given by

$$p(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \alpha_0) = p(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{\beta}, \alpha_0)p(\boldsymbol{X}|\boldsymbol{\alpha})p(\boldsymbol{\alpha})p(\boldsymbol{\beta})p(\alpha_0) \tag{3.78}$$

From Equation (3.74), we can show that the posterior distribution of $\boldsymbol{X}$ is a complex Gaussian:

$$p(\boldsymbol{X}|\boldsymbol{Y}, \alpha_0, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \prod_{l=1}^{L} CN(\boldsymbol{x_l}|\boldsymbol{\mu_l}, \boldsymbol{\Sigma}) \tag{3.79}$$

where $\boldsymbol{\mu_l} = \alpha_0 \boldsymbol{\Sigma} \boldsymbol{\Phi}^H \boldsymbol{y_l}$ for $l = 1, 2, ..., L$ and $\boldsymbol{\Sigma} = (\alpha_0 \boldsymbol{\Phi}^H \boldsymbol{\Phi} + \boldsymbol{\Lambda}^{-1})^{-1}$. We need to estimate the parameters $\boldsymbol{\alpha}, \boldsymbol{\beta}, \alpha_0$ to calculate the mean and covariance of the above posterior distribution. An expectation-maximization algorithm is used to maximize the function $\mathrm{E}\{\log p(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \alpha_0)\}$ where $\mathrm{E}\{.\}$ denotes expectation with respect to the posterior of $\boldsymbol{X}$ given in Equation (3.79). The parameters $\boldsymbol{\alpha}, \boldsymbol{\beta}, \alpha_0$ are updated so that the above expectation function is maximized [55].

The steps of the OGSBI algorithm as follows. First, the parameters $\boldsymbol{\alpha}, \boldsymbol{\beta}, \alpha_0$ are initialized. The mean and covariance of the posterior of $\boldsymbol{X}$ are calculated using the current values of the parameters. Then, the parameters are updated so that the function $\mathrm{E}\{\log p(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \alpha_0)\}$ is maximized. This process is repeated until convergence.

3.6 Gridless Sparse Methods

Gridless sparse methods remove the grid concept in searching the solution from infinite dimensional parameter sample set. To do so, a new sparse metric named as atomic norm is introduced. Consider the columns of the dictionary matrix $\boldsymbol{A}$ as atoms from an atomic set. The reason for the reconstruction errors arising from the grid mismatches is finite dimensionality of the set of atoms. Atomic norm concept comes with an infinite dimensional atomic set $\mathcal{A}$. Atomic norm for an n-dimensional vector z is defined as the Minkowski functional of convex hull of the atomic set $conv(\mathcal{A})$:

$$\begin{aligned} \|\boldsymbol{z}\|_{\mathcal{A}} =& \text{inf}\{t \geq 0 : \boldsymbol{z} \in tconv(\mathcal{A})\} \\ =& \text{inf}\left\{\sum_i x_i : \boldsymbol{z} = \sum_i x_i \boldsymbol{a}_i, x_i \geq 0, \boldsymbol{a}_i \in \mathcal{A}\right\} \end{aligned} \tag{3.80}$$

From the above definition, it is seen that the atomic norm includes the ℓ_1 norm as a special case but it is quite general where $\mathcal{A}$ can be an infinite dimensional atomic set. The atomic norm can be defined for matrices as well. The atomic norm of the matrix $\boldsymbol{Z}$ is given by

$$\begin{aligned} \|\boldsymbol{Z}\|_{\mathcal{A}} =& \text{inf}\{t \geq 0 : \boldsymbol{Z} \in tconv(\mathcal{A})\} \\ =& \text{inf}\left\{\sum_i \|\boldsymbol{x_i}\|_2 : \boldsymbol{Z} = \sum_i \boldsymbol{a}_i \boldsymbol{x_i}, \boldsymbol{a}_i \in \mathcal{A}\right\} \end{aligned} \tag{3.81}$$

In fact, $\|\boldsymbol{Z}\|_{\mathcal{A}}$ is the continuous-valued counterpart of the $\ell_{2,1}$-norm. ℓ_1-norm and $\ell_{2,1}$-norm are used as regularization functions for single measurement vector and multi-

ple measurement vectors, respectively in on-grid sparse methods. In gridless sparse methods, instead of the discrete ℓ_1-norm and $\ell_{2,1}$-norms, the atomic norm is used as regularization function. Among the gridless sparse methods, atomic norm minimization approach is the most studied method and it has strong theoretical guarantees [56]. Therefore, we discuss the atomic norm minimization approach in the following section.

3.6.1 SMV Case

3.6.1.1 Atomic Norm Minimization Approach

Recall that the sparse signal representation of SMV parametric reconstruction problems in Equation (3.8). This representation can be written as follows:

$$\boldsymbol{y} = \boldsymbol{z} + \boldsymbol{w}, \text{ and } \boldsymbol{z} = \boldsymbol{A}\boldsymbol{x} \tag{3.82}$$

where $\|\boldsymbol{x}\|_0 = K$ and $\boldsymbol{z}$ denotes the noiseless signal. For sparsity-based methods, we solve the following regularized least-squares problem:

$$\min_{\boldsymbol{z}} \quad \frac{1}{2}||\boldsymbol{y} - \boldsymbol{z}||_2^2 + \lambda\mathcal{R}(\boldsymbol{z}) \tag{3.83}$$

where λ is a nonnegative regularization parameter and the function $\mathcal{R}(\boldsymbol{z})$ is a sparse metric. Atomic norm minimization method solves the optimization problem in Equation (3.83) introducing the atomic norm as a sparse metric. The new optimization problem is obtained as follows:

$$\min_{\boldsymbol{z}} \quad \frac{1}{2}||\boldsymbol{y} - \boldsymbol{z}||_2^2 + \lambda\|\boldsymbol{z}\|_{\mathcal{A}} \tag{3.84}$$

However, the calculation of the atomic norm from the definition is unclear and we need to find an equivalent formulation to the atomic norm to solve the optimization problem Equation (3.84). Later the definition of the atomic norm, semi-definite characterization of the atomic norm minimization is introduced in the literature [57]. The atomic norm of $\boldsymbol{z}$ is equal to the optimal value of the following semi-definite programming problem:

$$\|\boldsymbol{z}\|_{\mathcal{A}} = \min_{t,\boldsymbol{u}} \quad \frac{1}{2}t + \frac{1}{2}u_1 \quad \text{s.t.} \quad \begin{bmatrix} t & \boldsymbol{z}^H \\ \boldsymbol{z} & \text{Toep}(\boldsymbol{u}) \end{bmatrix} \geq \boldsymbol{0} \tag{3.85}$$

where Toep($\boldsymbol{u}$) denotes the Toeplitz matrix having the vector $\boldsymbol{u}$ as its first column, u_1 is the first entry of the vector $\boldsymbol{u}$ and t is a free variable. The computationally efficient semi-definite programming formulation that solves SMV parametric reconstruction problem is formed putting Equation (3.85) into Equation (3.84).

$$\min_{\boldsymbol{z},\boldsymbol{u},t} \quad \frac{1}{2}||\boldsymbol{y}-\boldsymbol{z}||_2^2+\frac{\lambda}{2}(t+u_1) \quad \text{s.t.} \quad \begin{bmatrix} t & \boldsymbol{z}^H \\ \boldsymbol{z} & \text{Toep}(\boldsymbol{u}) \end{bmatrix} \geq \boldsymbol{0} \tag{3.86}$$

The SDP formulation in Equation (3.86) solves the SMV parametric reconstruction problem. However, the selection of the regularization parameter λ is important to efficient optimization. Under independent and identically distributed Gaussian noise, [57] shows that if we set $\lambda \approx \sqrt{T \log T\sigma}$ in Equation (3.86), the parameters and the sparse vector can be stably estimated.

3.6.2 MMV Case

3.6.2.1 Atomic Norm Minimization Approach

The atomic norm minimization approach can be easily extended to the multiple measurement vectors case. Similar to $\|\boldsymbol{z}\|_{\mathcal{A}}$, the atomic norm $\|\boldsymbol{Z}\|_{\mathcal{A}}$ has semi-definite characterization to provide efficient computation [58]:

$$\|\boldsymbol{Z}\|_{\mathcal{A}} = \min_{\boldsymbol{T},\boldsymbol{u}} \quad \frac{1}{2}\left[Tr(\boldsymbol{T})+Tr(\text{Toep}(\boldsymbol{u}))\right] \quad \text{s.t.} \quad \begin{bmatrix} \boldsymbol{T} & \boldsymbol{Z}^H \\ \boldsymbol{Z} & \text{Toep}(\boldsymbol{u}) \end{bmatrix} \geq \boldsymbol{0} \tag{3.87}$$

where $Tr(\boldsymbol{A})$ is the trace of the matrix $\boldsymbol{A}$ and $\boldsymbol{T}$ is a free matrix.

For multiple measurement vectors, sparsity-based methods solve the following optimization problem in general:

$$\min_{\boldsymbol{Z}} \quad \frac{1}{2}||\boldsymbol{Y}-\boldsymbol{Z}||_2^2+\lambda\mathcal{R}(\boldsymbol{Z}) \tag{3.88}$$

Atomic norm minimization in case of multiple measurement vectors put the atomic norm $\|\boldsymbol{Z}\|_{\mathcal{A}}$ into Equation (3.88). The resulting optimization problem using Equation (3.87) and (3.88) is given by

$$\min_{\boldsymbol{T},\boldsymbol{z},\boldsymbol{u}} \quad \frac{1}{2}||\boldsymbol{Y}-\boldsymbol{Z}||_F^2+\frac{\lambda}{2}\left[Tr(\boldsymbol{T})+Tr(\text{Toep}(\boldsymbol{u}))\right] \quad \text{s.t.} \quad \begin{bmatrix} \boldsymbol{T} & \boldsymbol{Z}^H \\ \boldsymbol{Z} & \text{Toep}(\boldsymbol{u}) \end{bmatrix} \geq \boldsymbol{0} \tag{3.89}$$

[58] shows that we can stably estimate the parameters and the sparse matrix with the choice of $\lambda \approx \sqrt{T(L + \log T + \sqrt{2L \log T})}\sigma$ in Equation (3.89).

The SDP formulations for both SMV and MMV case can be efficiently optimized via interior point algorithms [59]. There are different solvers for solving SDP, but the computational cost of these solvers are quite high. As an example, SDPT3 is an interior point algorithm, and the computational complexity of ANM with SDPT3 solver becomes $\mathcal{O}((T + L^2)^2 + (T + L)^{2.5})$ [60].

CHAPTER 4

PERFORMANCE COMPARISON FOR DIFFERENT METHODS

In this chapter, the performance of classical and sparsity-based methods are compared in direction-of-arrival and parameterized source separation applications. The focus of the first section of this chapter is to compare the performances of classical and sparsity-based methods in direction-of-arrival estimation. However, we focus on the effect of the discretization and the performance comparison of sparsity-based methods on the parameterized source separation in the second section. The reason is that there are two different parameters to be estimated in the parameterized source separation, and this causes the discretization issues.

4.1 Direction-of-Arrival (DOA) Estimation Problem

In this section, some of the methods that were presented in Chapters 2 and 3 are evaluated on direction-of-arrival estimation problem by several simulations. The simulation scenarios are selected so that both real-life conditions are realized, and the important aspects of each method are stated.

Recall that instead of the number of measurements, the snapshot is used in array processing literature. The single snapshot DOA estimation problem is an instance of SMV parametric reconstruction problems, whereas the multiple snapshots DOA estimation problem is an instance of MMV parametric reconstruction problems. We evaluate the performance of methods on the single snapshot and multiple snapshots DOA estimation problems in the following sections.

We introduce two **performance metrics** to measure the reconstruction accuracy of the methods on the DOA estimation problem. These are **root-mean-square-error**

(**RMSE**) and the **success rate** (**SR**). Moreover, the run-times of methods are used for comparison purposes.

RMSE for DOA estimation problem is defined as follows:

$$\text{RMSE} = \sqrt{\frac{1}{NC}\sum_{c=1}^{C}\sum_{n=1}^{N}(\theta_n - \hat{\theta}_{n,c})^2} \tag{4.1}$$

where C is the number of Monte Carlo trials, N is the number of sources, θ_n is the true DOA and $\hat{\theta}_{k,t}$ is the estimate of DOA of the n$^{\text{th}}$ source in trial c.

The other performance metric is the **success rate**, and it is calculated as the ratio of the number of successful trials to the total number of trials. If the angle difference between true and the reconstructed DOA is smaller than or equal to 1°, then the trial is considered successful.

In this thesis, only 1D setting is considered for the DOA estimation problem. That is, the elevation angle of the sources are selected as 90°. The estimated arrival angles are in the azimuth plane. The placement of the sensor array for DOA estimation problem is given in Figure 4.1. An 8-element uniform linear array is used in the problem. The wavelength of the impinging signals is set to $1\ m$ and the distance between two consecutive sensors is set to half a wavelength, i.e., $\Delta = 0.5\ m$.

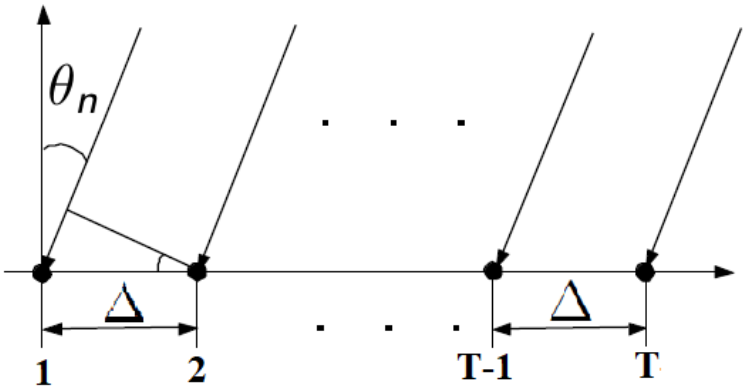

Figure 4.1: The uniform linear array

The noise samples are selected from i.i.d. complex Gaussian distribution, $w_i \sim \mathcal{CN}(0, \sigma^2)$, $i = 1, 2, ..., T$ where w_i is the i$^{\text{th}}$ sample of the noise and T is the number of sensors in the problem. SNR is defined as ten times of the $\log$-ratio of

the mean energy of the noiseless signal and the noise variance, σ^2. In case of single snapshot DOA estimation problem, SNR is given by

$$\text{SNR} = 10\log_{10}\frac{\|\boldsymbol{A}(\boldsymbol{\theta})\boldsymbol{s}\|_2^2/T}{\sigma^2} \tag{4.2}$$

In case of multiple snapshot DOA estimation problem, SNR is given by

$$\text{SNR} = 10\log_{10}\frac{\|\boldsymbol{A}(\boldsymbol{\theta})\boldsymbol{s}\|_2^2/(T\times L)}{\sigma^2} \tag{4.3}$$

4.1.1 Single Snapshot DOA Estimation

In many practical applications, there are multiple numbers of snapshots available. However, a small number of snapshots are available due to physical constraints in some practical applications. For example, mostly, a single snapshot is available due to the limit of the speed of the sound in sonar processing [61]. Similarly, only a minimal number of snapshots are available in automotive radar systems [62]. In this section, the performance of the methods that deal with the SMV case is evaluated for a single snapshot DOA estimation problem. Some of the methods that are presented in Chapters 2 and 3 require multiple measurements, i.e., multiple snapshots to obtain reasonable results. These methods are not covered in this section. The methods that are evaluated in this section are BPDN, FOCUSS, OGSBI, SURE-IR, and ANM. Although OGSBI is an MMV method, we can apply this method to a single snapshot DOA estimation problem because the algorithm steps can easily be reduced so that it can also be applied for an SMV model. Moreover, OMP could be included in this comparison as well; but it is well-known that OMP will not outperform BPDN in terms of accuracy and its computational efficiency does not provide a significant advantage over BPDN in the SMV case.

Recall that the single snapshot matrix-vector multiplication model for DOA estimation problem is given by

$$\boldsymbol{y} = \boldsymbol{A}(\boldsymbol{\theta})\boldsymbol{s} + \boldsymbol{w} \tag{4.4}$$

When conducting the single snapshot DOA estimation experiment, each true source signal s_i is selected from zero mean unit variance complex Gaussian distribution.

The sparse linear system representation of this parametric model is given by

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{w} \tag{4.5}$$

where $\boldsymbol{A}$ is an over-determined matrix obtained by discretizing the parameter space and $||\boldsymbol{x}||_0 = N$. In single snapshot DOA estimation problem, the interval of arrival angles is $[-90°, 90°]$ in it is sampled with $1°$. Therefore, the candidate arrival angle set is $[-90, -89, ..., 89, 90]$ for grid based sparse methods. Also, $\boldsymbol{x}$ is zero-padded version of $\boldsymbol{s}$ in Equation (4.4).

Below, three simulation scenarios are determined according to the placement of the source signals, the correlation between them, and the number of the sources. For each simulation scenario, the true arrival angles of the sources are selected both on the grid points and out of the grid points to see the performance of grid-based and off-grid sparse methods. The results are the average reconstructions of 100 realizations. Simulation results for these scenarios are shown in Figure 4.2 through Figure 4.13. Simulation results are shown as a function of SNR.

4.1.1.1 Scenario-I: Two Well-Separated Uncorrelated Sources

In this scenario, there are two far-field uncorrelated sources impinging upon the uniform linear array. The difference of the arrival angles of the sources is selected as $30°$ to separate the sources well. The sources are generated so that they are uncorrelated, i.e. each source signal is independently selected from zero mean complex Gaussian distribution:

$$\boldsymbol{s} = \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} \tag{4.6}$$

where $s_i \sim CN(0,1)$, $i = 1, 2$ with $CN(0,1)$ being a zero mean unit variance complex Gaussian distribution.

For Figures 4.2 and 4.3, the sources are located at $10°$ and $40°$. For Figures 4.4 and 4.5, the sources are located at $10.5°$ and $40.5°$. The results of this scenario provide a baseline for the performances of the methods.

It is easily seen that from Figures 4.2, 4.3, 4.4 and 4.5, OGSBI method fails in this scenario. Remember that the OGSBI is a Bayesian method, and it uses unknown

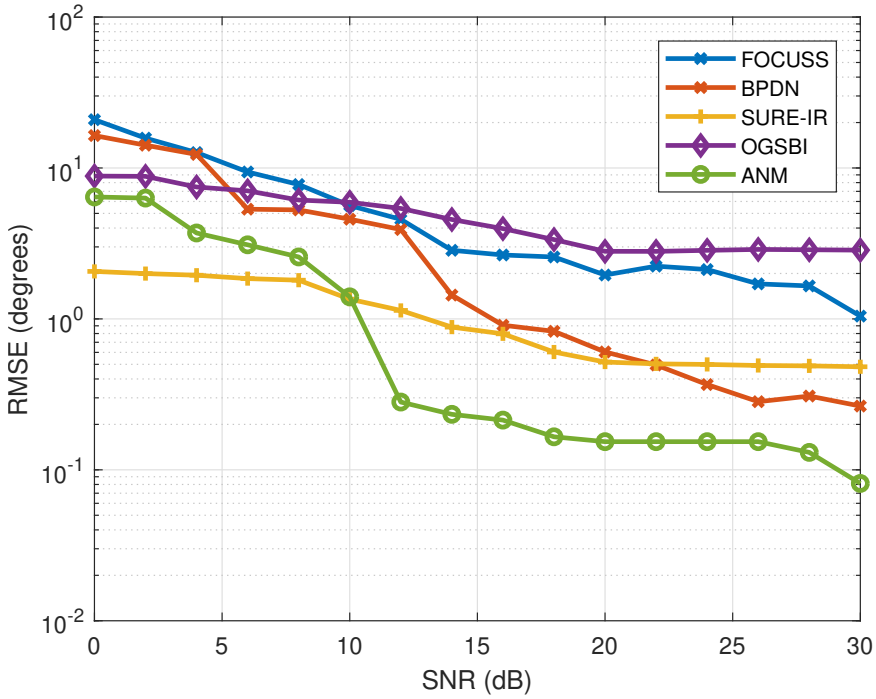

Figure 4.2: RMSE when two uncorrelated sources are well-separated and on the grid, $L = 1$

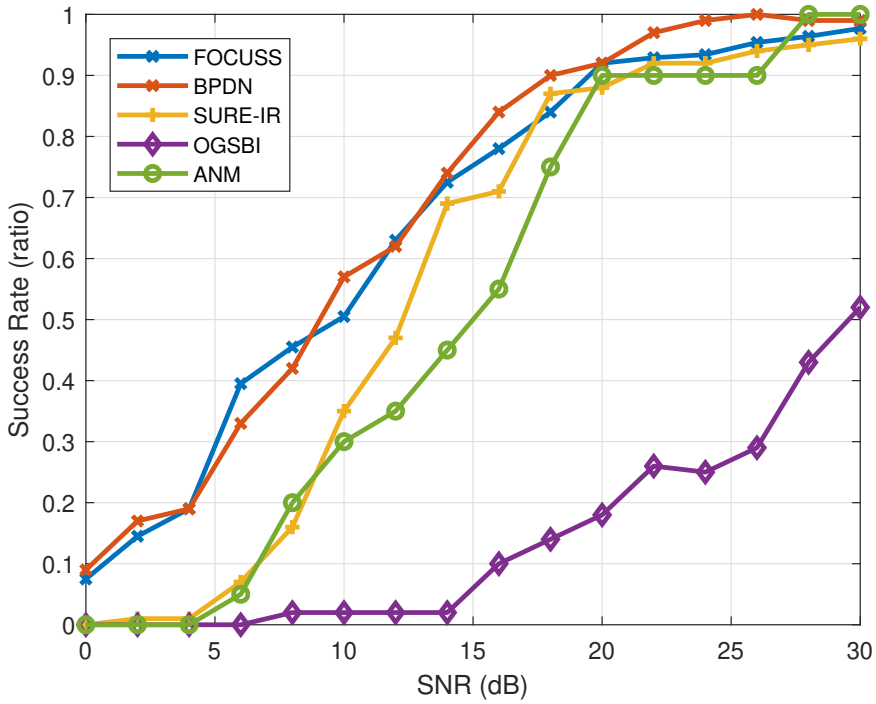

Figure 4.3: Success rate when two uncorrelated sources are well-separated and on the grid, $L = 1$

hyper-parameters $\boldsymbol{\alpha}, \boldsymbol{\beta}, \alpha_0$ that need to be estimated in each iteration. Consequently, the total number of parameters to be estimated increases, and there should be a certain number of measurements to estimate the parameters accurately. As a result, the single snapshot measurement is not enough for the OGSBI method in this scenario.

FOCUSS and BPDN are two grid-based sparse methods. From Figures 4.2 and 4.3, we can say that BPDN outperforms FOCUSS for all SNR range when the sources are located on the grid. From Figures 4.4 and 4.5, the performance of BPDN is nearly the same when the sources are located out of the grid. However, the performance of FOCUSS decreases when the sources are located out of the grid. The reason is that the sparsity is explicitly involved in the objective function of BPDN, whereas FOCUSS indirectly enforces sparsity.

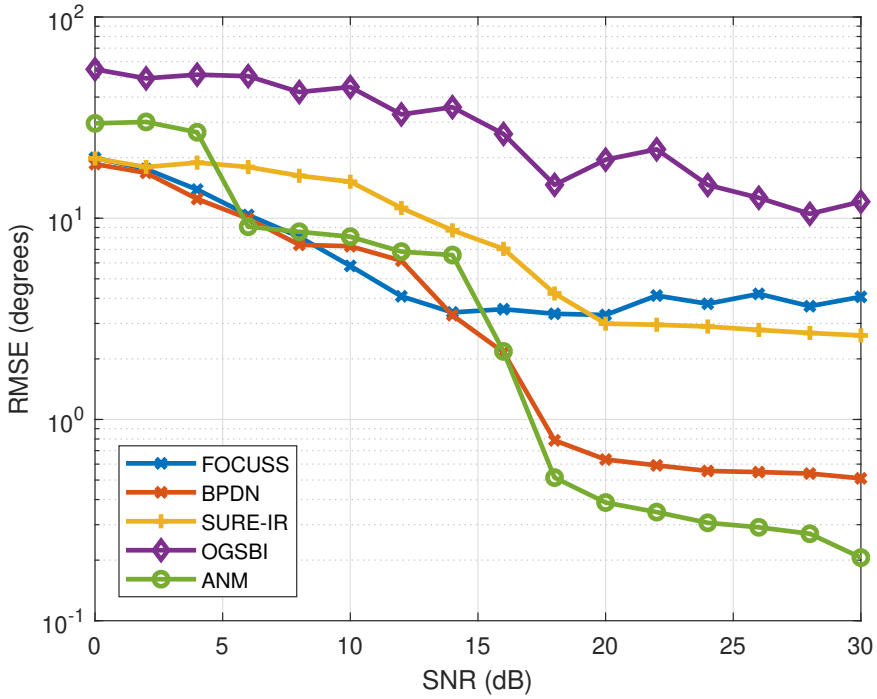

Figure 4.4: RMSE when two uncorrelated sources are well-separated and out of the grid, $L = 1$

From Figures 4.2 and 4.4, SURE-IR does not perform as desired. Its performance decreases when the sources are located out of the grid. SURE-IR is an off-grid sparse method that uses alternating minimization. The method iteratively estimates the sparse vector and updates the dictionary. This two-stage algorithm does not guarantee to obtain the optimum value of the objective function at the end of iterations.

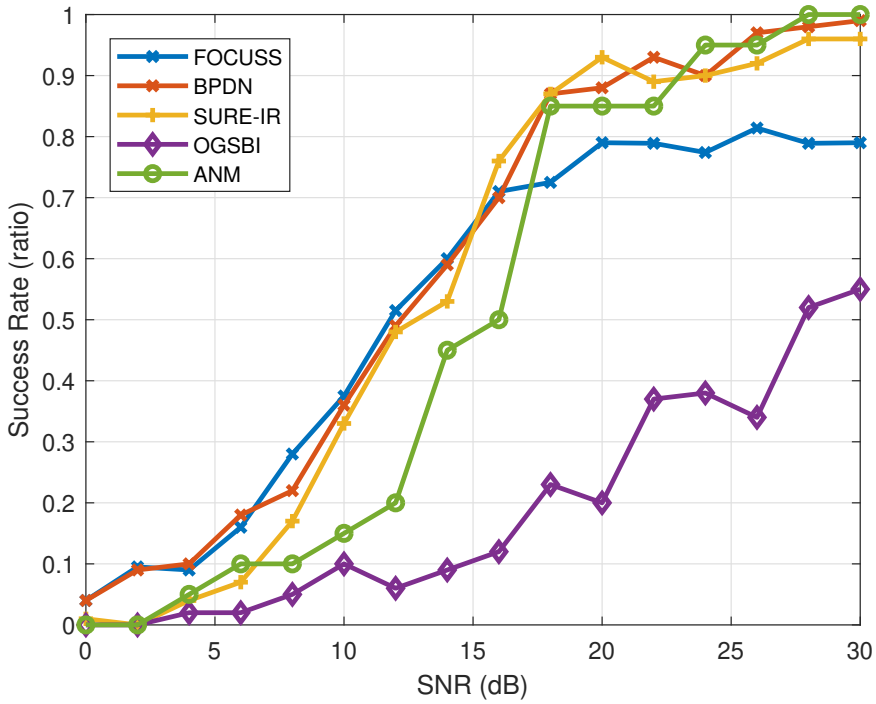

Figure 4.5: Success rate when two uncorrelated sources are well-separated and out of the grid, $L = 1$

Therefore, the algorithm may not perform satisfactorily.

As a gridless sparse method, ANM outperforms other methods in high SNR range for both sources are located on the grid or out of the grid in this scenario. It is reasonable since ANM does not use a discretization scheme; it operates on a continuous-valued parameter domain. However, ANM does not give satisfactory results in low SNR range [57].

When the SNR is smaller than 10 dB, any methods perform satisfactorily because of both noise and the low number of measurements. This SNR range is discarded when discussing the performance of the methods.

Table 4.1 shows the average run times of methods for this scenario. These values are proportional to the computational complexities. ANM is the most accurate method among the sparse methods, but it suffers from high computational complexity. Similar to ANM, the off-grid sparse methods, SURE-IR and OGSBI, also suffer from the computational complexity because they search the off-grid points. Note that the relation between the run times of methods is approximately the same for the follow-

Table 4.1: Average run times of methods for Scenario-I

Method	**Average Run time (s)**
BPDN	0.1136
FOCUSS	0.0126
SURE-IR	1.2148
OGSBI	1.1150
ANM	5.8121

ing scenarios. For this reason, we do not include the run times of methods in the following scenarios.

4.1.1.2 Scenario-II: Two Closely-Spaced Uncorrelated Sources

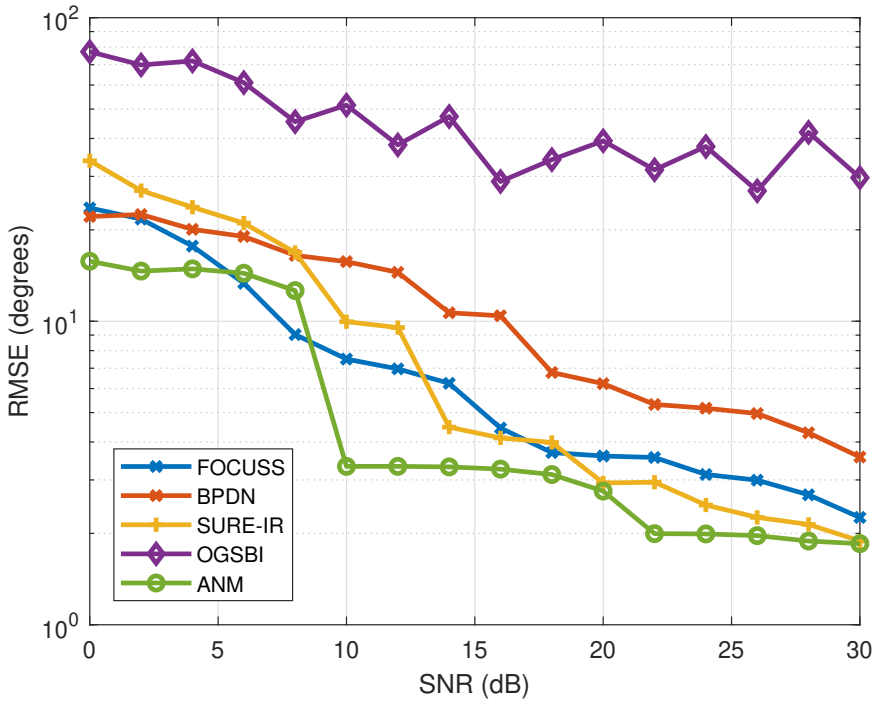

Figure 4.6: RMSE when two uncorrelated sources are closely-spaced and on the grid, $L = 1$

Resolving two closely-spaced sources is an essential issue in the DOA estimation problem. The ability to resolve two closely-spaced sources (separated by less than a

beamwidth) is called super-resolution [23].

We investigate the performance of the methods for two closely-spaced sources. To do this, the difference of the arrival angles of the sources is selected as $5°$. The sources are generated so that they are uncorrelated as in 4.1.1.1. For Figures 4.6 and 4.7, the sources are located at $10°$ and $15°$. For Figures 4.8 and 4.9, the sources are located at $10.5°$ and $15.5°$.

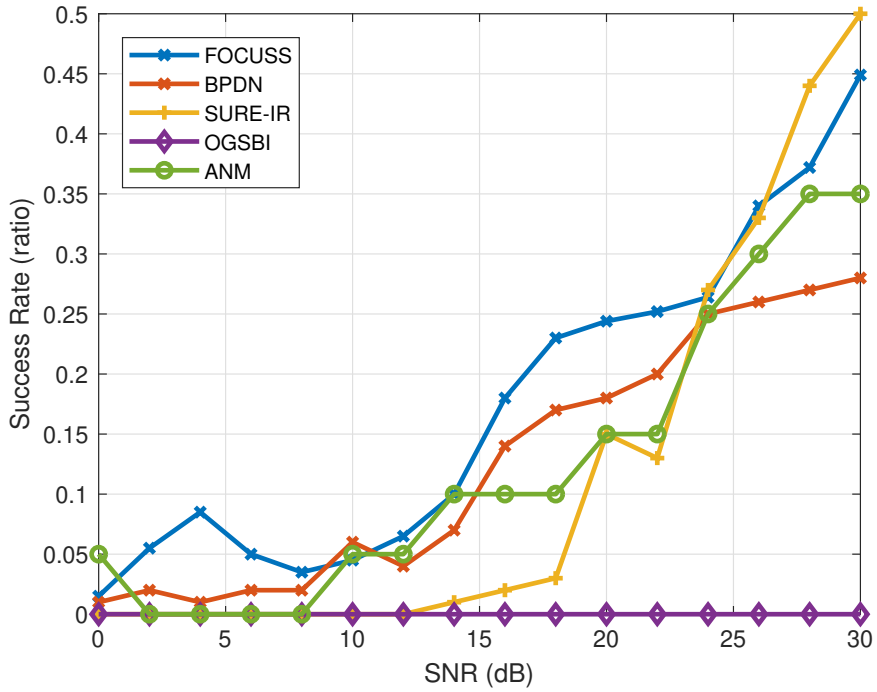

Figure 4.7: Success rate when two uncorrelated sources are closely-spaced and on the grid, $L = 1$

As it is seen from Figures 4.6 and 4.8, the performance of all methods decreases with closely-spaced sources. None of the methods gives an RMSE value smaller than $1°$. Therefore, none of the methods has a success rate close to 1 as can be seen in Figure 4.7 and 4.9. OGSBI method fails in this scenario because of the same reason as in 4.1.1.1.

From figure 4.6, it is seen that the FOCUSS method performs better than BPDN for the whole SNR range. BPDN is sensitive to closely-spaced sources case more than FOCUSS is. However, the performance of FOCUSS decreases when the sources are located out of the grid. It is expected because FOCUSS estimates the arrival angles

that are on the grid.

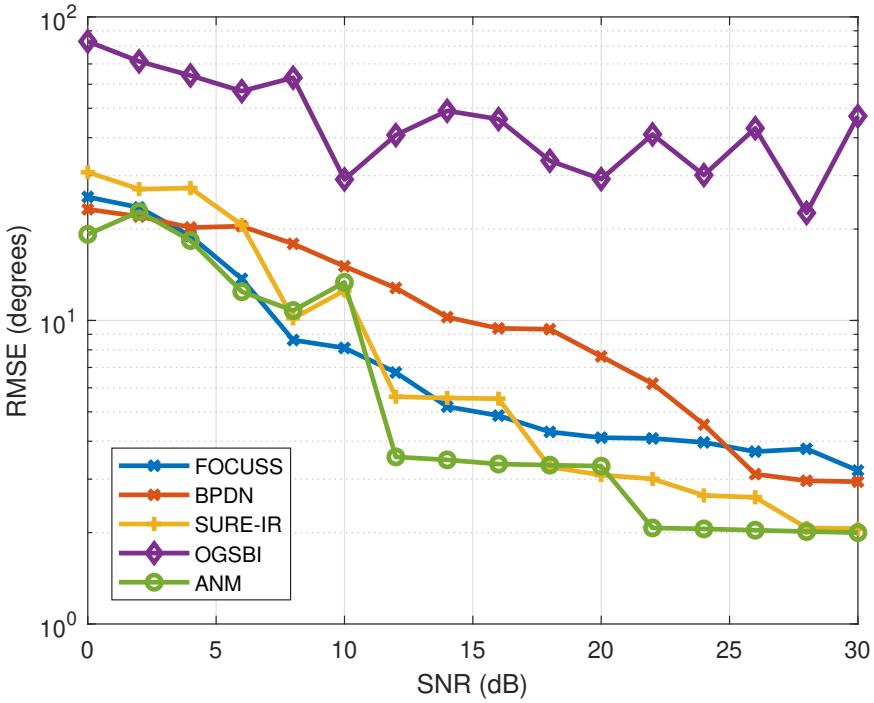

Figure 4.8: RMSE when two uncorrelated sources are closely-spaced and out of the grid, $L = 1$

It is evident from Figure 4.6 and 4.8, having closely-spaced sources decreases the performance of SURE-IR less than that of FOCUSS and BPDN. It is expected because the columns of the over-determined matrix $\boldsymbol{A}$ in Equation (4.5) that corresponds to the closely-spaced sources are similar to each other. This similarity complicates to resolve two sources for on-grid methods.

As in 4.1.1.1, the ANM method outperforms the other methods in high SNR range for both sources are located on the grid or out of the grid, as can be seen in Figure 4.6 and 4.8, respectively. The reason is that ANM does not use a discretization scheme, unlike the other sparsity-based methods. Note that the success rate values are much smaller than 1, and these small success rate values are not reliable for comparison.

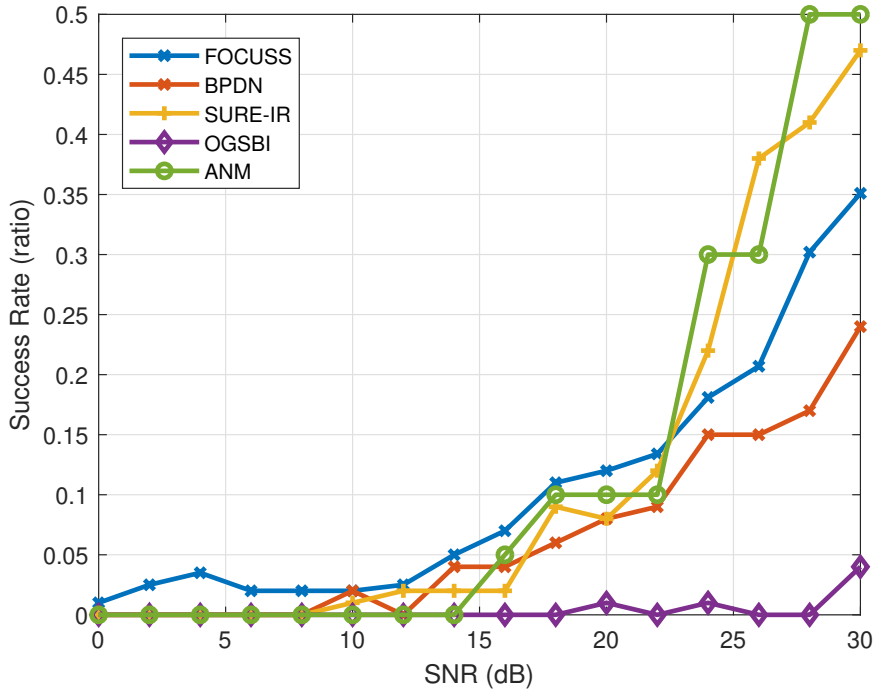

Figure 4.9: Success rate when two uncorrelated sources are closely-spaced and out of the grid, $L = 1$

4.1.1.3 Scenario-III: Two Well-Separated Correlated Sources

In previous scenarios, the correlation coefficient between the sources is selected as $\mu = 0$ so that the sources are uncorrelated. In this scenario, the correlation coefficient between the two source signals is selected as $\mu = 0.99$. The correlated sources can occur in practical applications due to multi-path effects [63]. For example, correlated signals are often encountered in smart jamming in radar systems and wireless communications [15].

Recall that the SMV parametric model for single-snapshot DOA estimation problem with uncorrelated sources as follows:

$$\boldsymbol{y} = \boldsymbol{A}(\boldsymbol{\theta})\boldsymbol{s} + \boldsymbol{n} \tag{4.7}$$

where the vector $\boldsymbol{s} = [s_1 \;\; s_2]^T$ and $s_i \sim CN(0,1)$, $i = 1, 2$. The model with correlated sources is given by

$$\boldsymbol{y} = \boldsymbol{A}(\boldsymbol{\theta})\hat{\boldsymbol{s}} + \boldsymbol{n} \tag{4.8}$$

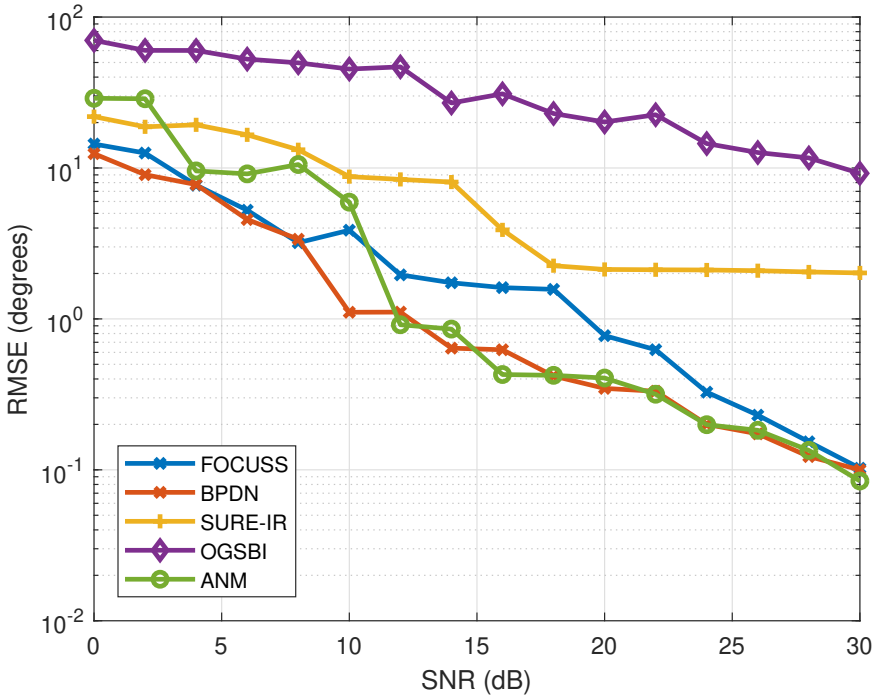

Figure 4.10: RMSE when two highly correlated sources are well-separated and on the grid, $L = 1$

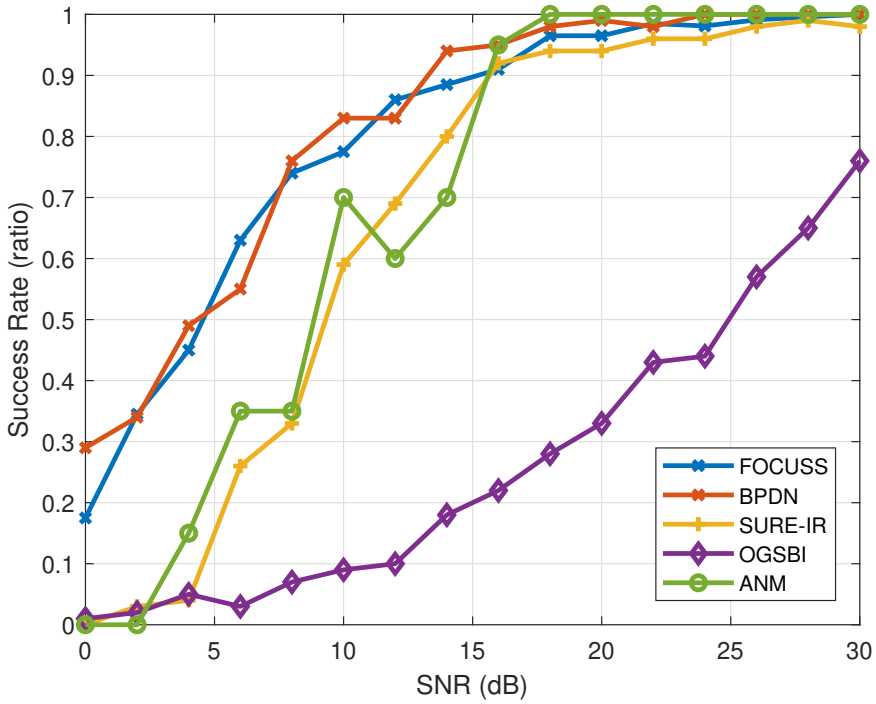

Figure 4.11: Success rate when two highly correlated sources are well-separated and on the grid, $L = 1$

where $\hat{\boldsymbol{s}} = \begin{bmatrix} s_1 \\ 0.99s_1 + \sqrt{1-0.99^2}s_2 \end{bmatrix}$

We investigate the performance of the methods for correlated sources. For Figures 4.10 and 4.11, the sources are located at $10°$ and $40°$. For Figures 4.12 and 4.13, the sources are located at $10.5°$ and $40.5°$. The only difference with 4.1.1.1 is the source correlation.

As it is seen from Figures 4.10, 4.11, 4.12 and 4.13, OGSBI method also fails in this scenario because the number of snapshot is not sufficient for this method.

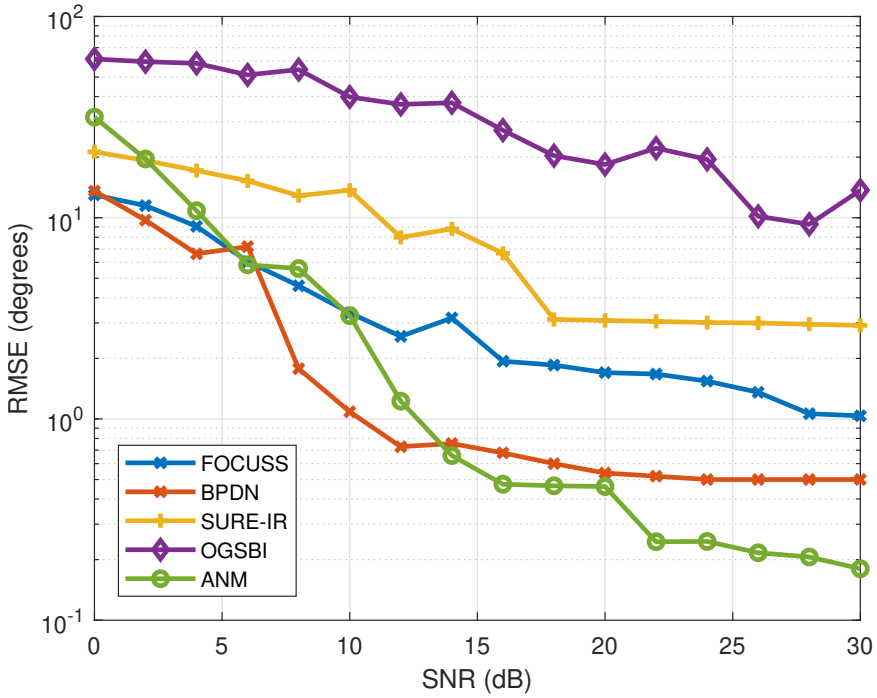

Figure 4.12: RMSE when two highly correlated sources are well-separated and out of the grid, $L = 1$

If we compare the Figures 4.2 and 4.10, we see that the source correlation does not degrade the performance of FOCUSS, BPDN, and ANM when the sources are located on the grid. Also, from Figures 4.4 and 4.12, it is seen that the performance of BPDN, FOCUSS and ANM are nearly the same for uncorrelated and correlated sources when the sources are located out of the grid. The reason is that these methods do not use information about the source correlation, unlike subspace-based methods.

From Figures 4.10 and 4.12, we see that SURE-IR shows poor performance in this

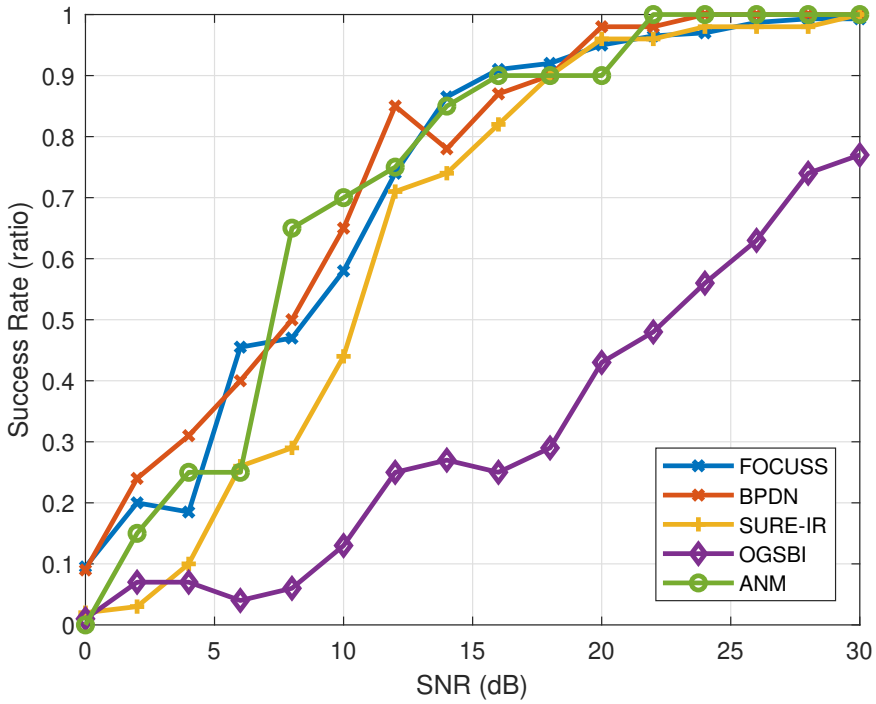

Figure 4.13: Success rate when two highly correlated sources are well-separated and out of the grid, $L = 1$

scenario as in 4.1.1.1. However, the success rate of SURE-IR for the high SNR range is over 0.9 when both sources are located on the grid and out of the grid from Figures 4.11 and 4.13. Although the success rate is high enough, RMSE values are approximately $2 - 3°$. That means the method shows poor performance, although it estimates the arrival angles accurately at most of the realizations. This behavior can be explained via the alternating structure of the method.

As a result, we have investigated the performance of several popular parametric reconstruction methods on the single snapshot DOA estimation problem. Three practically encountered scenarios have determined according to the location of sources and correlation between them. We have shown that the ANM method achieves the best estimation performance in high SNR range among the investigated methods, but it is not robust to noise. Other than this method, BPDN achieves excellent estimation performance when the sources are not closely-spaced. The performance of FOCUSS is useful when the sources are closely-spaced. We have seen that the SURE-IR method does not give consistent results because there is no theoretical guarantee for

this method. Also, the OGSBI method fails in this experiment since a single snapshot is not sufficient to accurately estimate the arrival angles.

4.1.2 Multiple Snapshot DOA Estimation

In this section, the performance of methods deals with the MMV parametric reconstruction problems is evaluated on multiple snapshots DOA estimation. All of the methods presented in Section 2 can be used to solve the MMV parametric reconstruction problems, including methods that are applied to the SMV problem. We can treat the MMV parametric reconstruction problems as L different SMV problems. Then, the methods that apply to the SMV parametric reconstruction problems can be applied to these L problems separately. However, these methods do not consider the joint information in the MMV problem, and this leads to poor performance. We decide not to evaluate these methods in multiple snapshot DOA estimation problem. The methods that are evaluated in this section are MUSIC, ESPRIT, ℓ_1-SVD, OGSBI, and ANM.

The classical methods require the knowledge of the number of unknown signals. On the other hand, sparse methods do not require this. In the multiple snapshot DOA estimation problem, the number of sources is treated as known for the classical methods including MUSIC and ESPRIT.

Recall that the multiple snapshots matrix-vector multiplication model for DOA estimation problem is given by

$$\boldsymbol{Y} = \boldsymbol{A}(\boldsymbol{\theta})\boldsymbol{S} + \boldsymbol{W} \tag{4.9}$$

where $\boldsymbol{S} = [\boldsymbol{s_1}, \boldsymbol{s_2}, ..., \boldsymbol{s_L}]$, $\boldsymbol{Y} = [\boldsymbol{y_1}, \boldsymbol{y_2}, ..., \boldsymbol{y_L}]$, $\boldsymbol{W} = [\boldsymbol{w_1}, \boldsymbol{w_2}, ..., \boldsymbol{w_L}]$ and $\boldsymbol{A}(\boldsymbol{\theta})$ is the same with $\boldsymbol{A}(\boldsymbol{\theta})$ in Equation (4.4). Here, the vectors $\boldsymbol{s_l}$, $\boldsymbol{y_l}$ and $\boldsymbol{w_l}$ is the same with the coefficient vector, measurement vector and noise vector in Equation (4.4), respectively. The MMV sparse linear system representation of this parametric model is given by

$$\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{X} + \boldsymbol{W}, \text{ for } l = 1, 2, .., L \tag{4.10}$$

where $\boldsymbol{X} = [\boldsymbol{x}_1, ..., \boldsymbol{x}_L]$, $||\boldsymbol{x}_l||_0 = N$ and each column of $\boldsymbol{X}$ is zero-padded version of respective column of $\boldsymbol{S}$ in Equation (4.9). When obtaining the overcomplete dic-

tionary matrix $\boldsymbol{A}$, the interval of arrival angles is $[-90°, 90°]$ and it is sampled with $1°$ as in single snapshot problem.

There are five simulation scenarios for multiple snapshot DOA estimation problem. The first three scenarios are the same as that of the single snapshot problem. The last two scenarios show the performance of the methods when the number of sources is increased to five, and different grid resolutions are considered.

4.1.2.1 Scenario-I: Two Well-Separated Uncorrelated Sources

Two uncorrelated far-field sources are placed so that the difference of arrival angles of these sources is $30°$. The sources are generated from zero-mean complex Gaussian distribution. The sources are also uncorrelated over different snapshots.

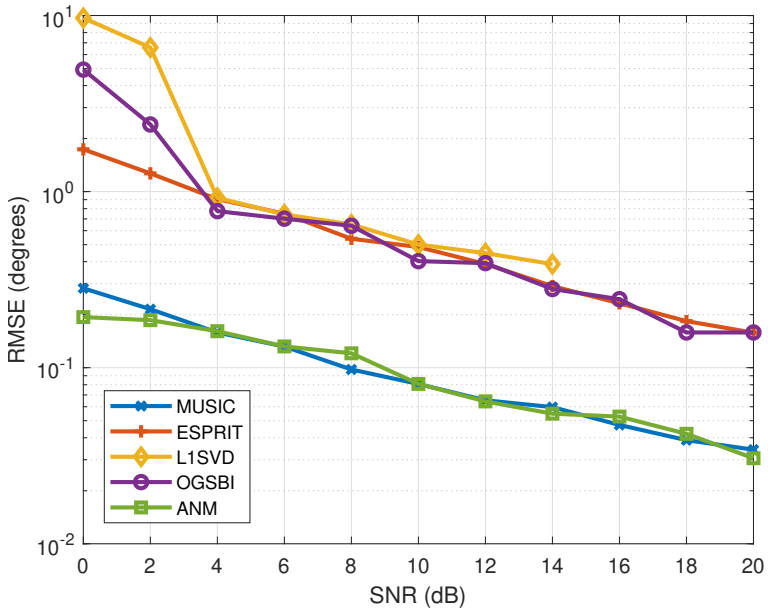

Figure 4.14: RMSE when two uncorrelated sources are well-separated and on the grid, $L = 100$

The sources are located at $10°$ and $40°$ for Figures 4.14 and 4.15. They are located at $10.5°$ and $40.5°$ for Figures 4.16 and 4.17. This scenario is the most basic scenario that provide baseline to other scenarios.

It is seen from figures 4.14, 4.15, 4.16 and 4.17 that MUSIC and ANM methods

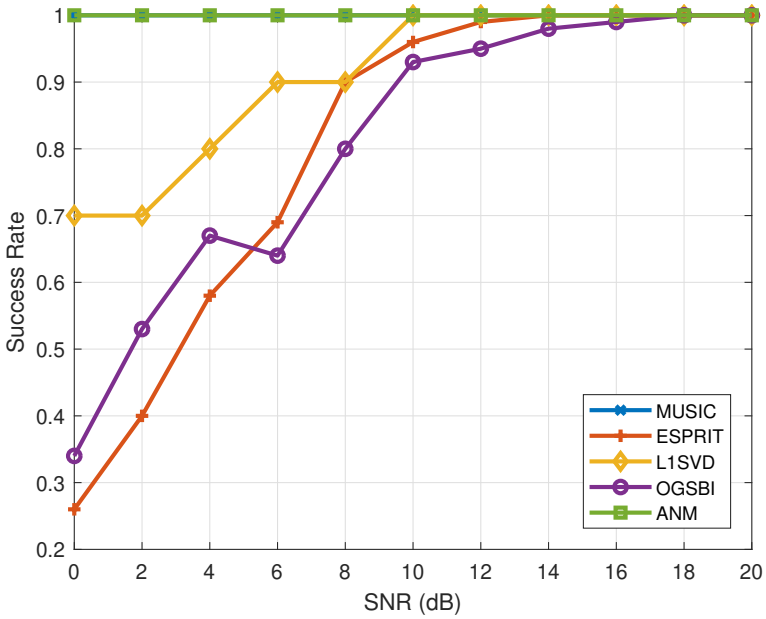

Figure 4.15: Success rate when two uncorrelated sources are well-separated and on the grid, $L = 100$

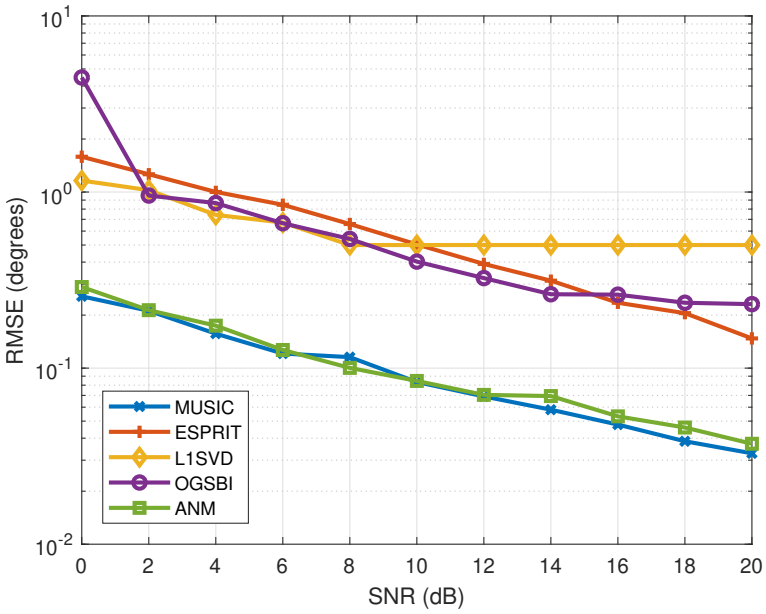

Figure 4.16: RMSE when two uncorrelated sources are well-separated and out of the grid, $L = 100$

outperform other methods for all SNR range except between 14 and 20 dB in 4.14. In fact, the success rate of these two methods are 1 for all SNR range. It shows that these methods perfectly reconstruct the angles in each simulation sample. Note that

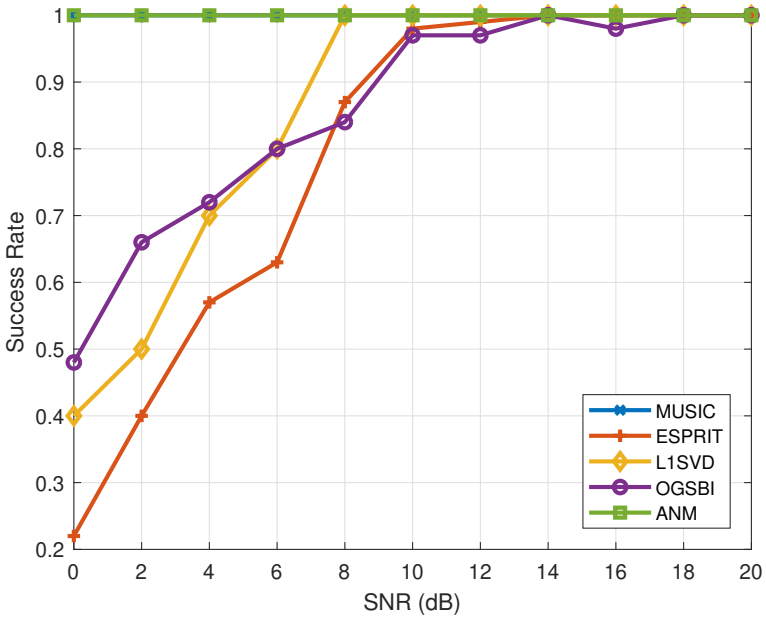

Figure 4.17: Success rate when two uncorrelated sources are well-separated and out of the grid, $L = 100$

there is no difference between Figure 4.14 and 4.16 for MUSIC and ANM methods because these methods are gridless.

In an earlier work [64], it was shown that MUSIC provides higher resolution and better accuracy than ESPRIT. If we compare the two subspace-based methods, MUSIC and ESPRIT, it is seen that the performance of MUSIC is better than that of ESPRIT for all SNR range.

Note that in Figure 4.14, the RMSE values are not shown for l1-SVD after 14 dB SNR. This is because the RMSE is zero for those high SNR cases and hence can not be shown in the semi-logarithmic plots used. It is expected because ℓ_1-SVD is an on-grid sparse method, and the sources are on the grid. When the noise is below a certain threshold, ℓ_1-SVD perfectly reconstructs the angles. On the other hand, the minimum RMSE value for ℓ_1-SVD is 0.5° when the sources are out of the grid according to Figure 4.16 because the sources are 0.5° out of the grid and ℓ_1-SVD estimates the angle that is on the closest grid point when the noise is below a certain threshold.

It is seen that from figures 4.14 and 4.16, OGSBI is capable to estimate the sources out of the grid unlike ℓ_1-SVD. In Figure 4.16, OGSBI gives an RMSE value smaller

Table 4.2: Average run times of methods for Scenario-I

Method	**Average Run time (s)**
MUSIC	0.0746
ESPRIT	0.0009
L1-SVD	6.3286
OGSBI	1.1371
ANM	144.3496

than 0.5° when SNR is larger than 10 dB. However, it is not as successful as MUSIC and ANM because it suffers from the underestimation of the noise variance [65].

Table 4.2 shows the average run times of each method in this scenario. According to this table, we can say that the sparsity-based methods have computational complexity more than that of the classical methods. The run time of the gridless sparse method, ANM, is too much than the other sparsity-based methods. Note that we will not include the run times for the other scenarios because the relation between the run times of methods is nearly the same for each scenario.

4.1.2.2 Scenario-II: Two Closely-Spaced Uncorrelated Sources

Two uncorrelated far-field sources are placed so that the difference of arrival angles of these sources is $2°$. The sources are generated from zero-mean complex Gaussian distribution. The sources are also uncorrelated over different snapshots.

The sources are located at $10°$ and $12°$ for Figures 4.18 and 4.19. They are located at $10.5°$ and $12.5°$ for Figures 4.20 and 4.21.

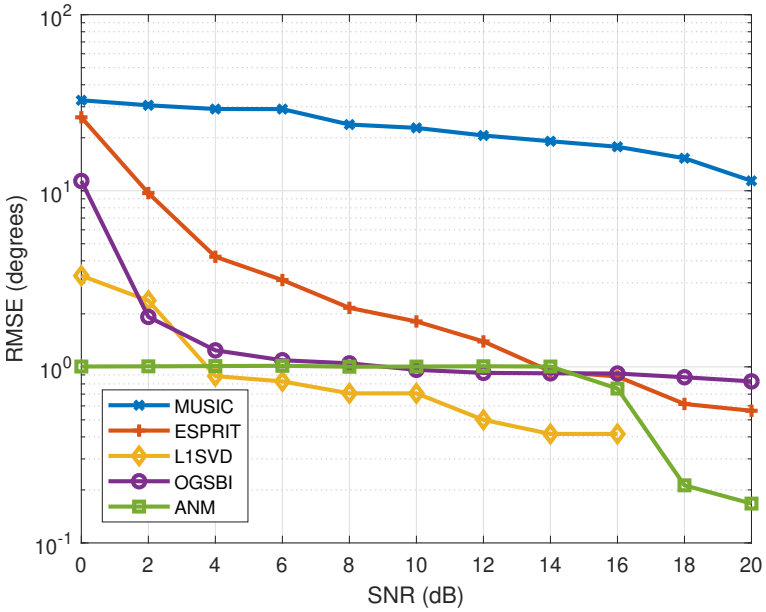

Figure 4.18: RMSE when two uncorrelated sources are closely-spaced and on the grid, $L = 100$

From Figures 4.18, 4.19,4.20 and 4.21, it is seen that MUSIC method fails. The reason is that MUSIC cannot resolve closely-spaced sources. However, ESPRIT is more robust to closely-spaced sources than MUSIC. This is seen from the figures.

Sparsity-based methods outperform the classical methods in most of the SNR range because being closely-spaced sources does not affect the sparsity-based methods as much as classical methods. Sparsity-based methods convert the nonlinear parameter estimation problem into a linear inverse problem.

From Figure 4.18, ℓ_1-SVD perfectly reconstruct the arrival angles when the SNR is larger than 16 dB. Note that the zero RMSE values after 16 dB are not shown because

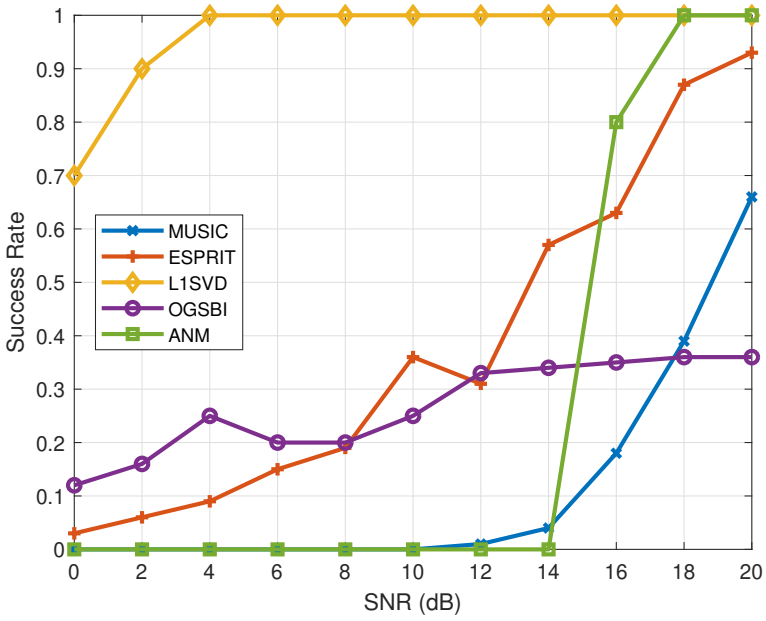

Figure 4.19: Success rate when two uncorrelated sources are closely-spaced and on the grid, $L = 100$

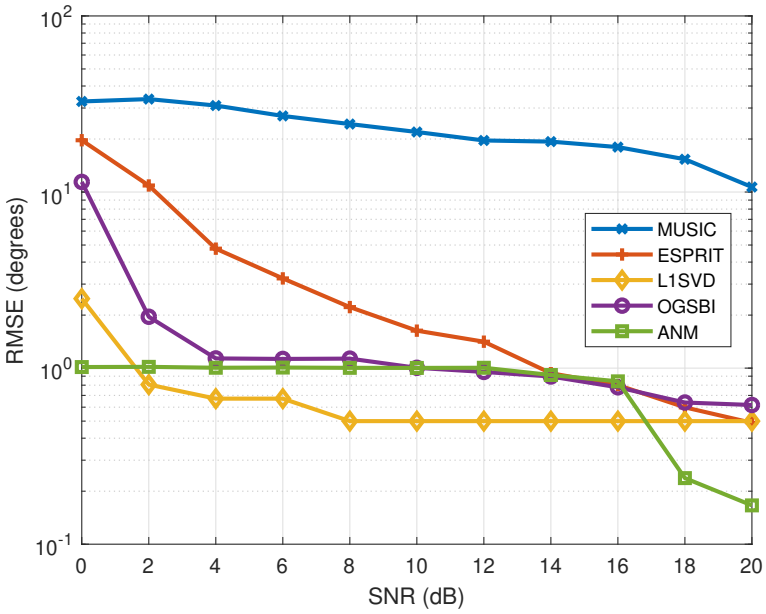

Figure 4.20: RMSE when two uncorrelated sources are closely-spaced and out of the grid, $L = 100$

they can not be shown in the semi-logarithmic plot. It is expected because the sources are on the grid. On the other hand, from Figure 4.20, ℓ_1-SVD gives an RMSE value of 0.5° when the SNR is larger than 8 dB. The method performs nearly the same

characteristic in Scenario-I and Scenario-II as expected.

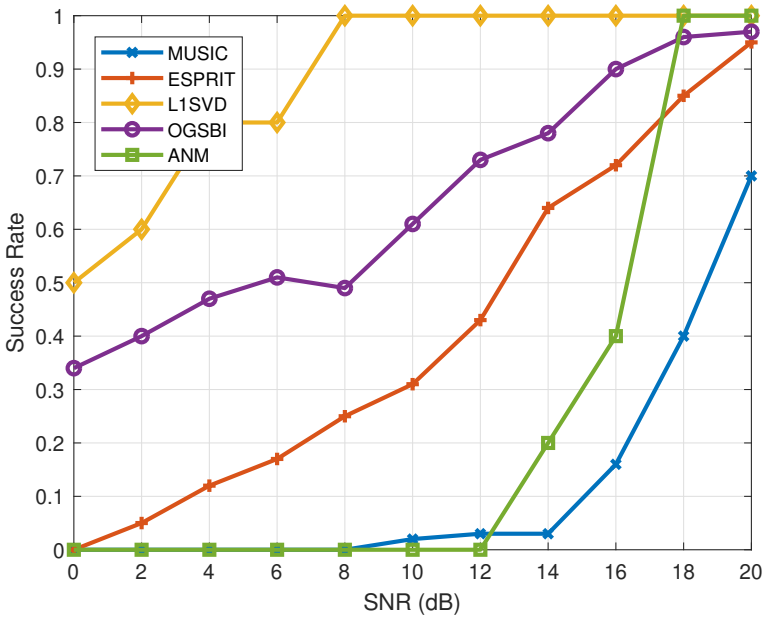

Figure 4.21: Success rate when two uncorrelated sources are closely-spaced and out of the grid, $L = 100$

From Figure 4.18, the best RMSE value for OGSBI is nearly 1°. In the case of off-grid sources, the performance of the method increases when the SNR is larger than 16 dB. RMSE value is nearly 0.6° in this range. In general, the OGSBI method shows poor performance in this scenario than the other sparsity-based methods.

The performance of the ANM method decreases with respect to the performance of ANM in Scenario-I. In most of the SNR range, ℓ_1-SVD outperforms ANM as it is seen in Figures 4.18 and 4.20. We infer that the ANM method does not give satisfactory results when the sources are closely-spaced. However, the effect of being closely-spaced sources to ANM is not as much as to classical methods. Note that, the most success rate values of ANM in Figure 4.19 and 4.21 are zero because the RMSE values of ANM is a bit larger than 1° in this SNR range and the corresponding success rate values to these RMSE values are zero.

4.1.2.3 Scenario-III: Two Well-Separated Correlated Sources

This scenario is the same as the Scenario-I except for the source correlation. In this scenario, the correlation coefficient between the two source signals is selected as $\mu = 0.99$. The correlation between the sources can occur in practical applications due to multi-path effects [63].

Classical methods fail in this scenario. This can be seen in Figures 4.22, 4.23, 4.24 and 4.25. The reason is that these methods assume that the sources are uncorrelated, and the algorithms run in this way. However, in the sparsity-based methods, not the signal subspaces but the powers of the source signals are used. Hence the performance of these methods is robust to the correlation between the sources [66].

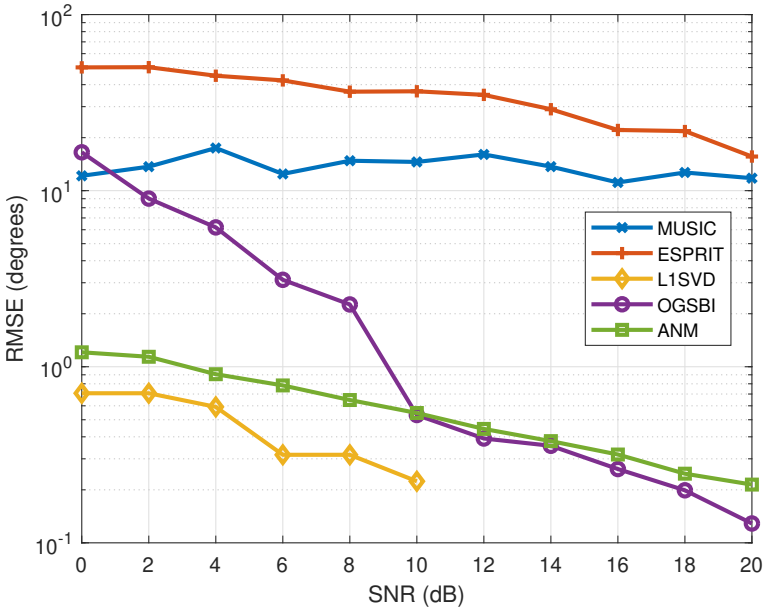

Figure 4.22: RMSE when two highly correlated sources are well-separated and on the grid, $L = 100$

Among the sparsity-based methods, source correlation affects the ANM method. Compared to the Scenario-I, the performance of the ANM method decreases. However, the success rate of the method is still 1 when the SNR is more than 6 dB, as it is seen in Figures 4.23 and 4.25.

Source correlation does not affect the behavior of ℓ_1-SVD as it is seen from Figures 4.22 and 4.24. For the sources on the grid, ℓ_1-SVD outperforms the other methods in

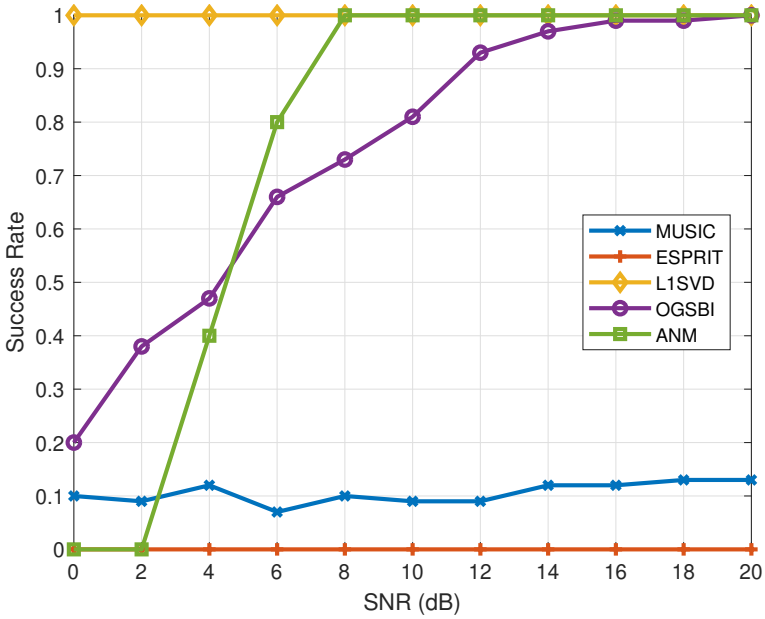

Figure 4.23: Success rate when two highly correlated sources are well-separated and on the grid, $L = 100$

all SNR range, and for the sources out of the grid, it outperforms the other methods when the SNR is smaller than 10 dB. ℓ_1-SVD obtains the best RMSE value which is $0.5°$ at 10 dB.

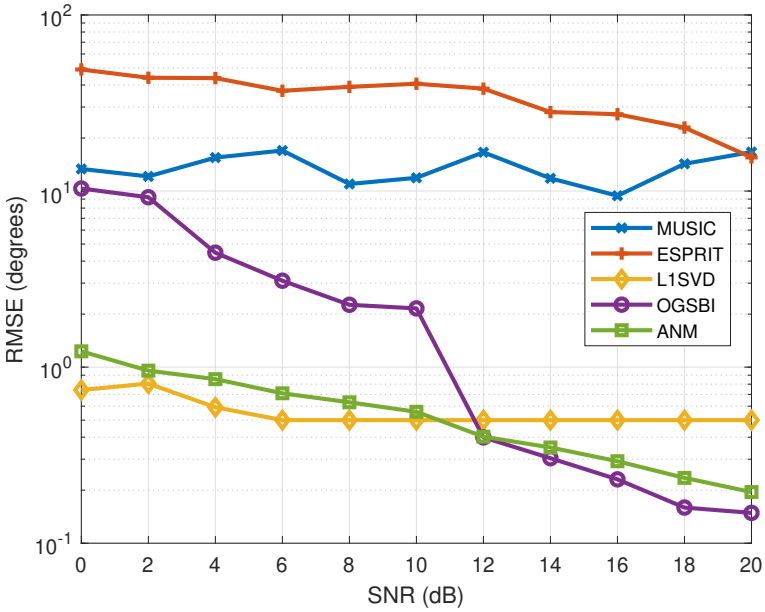

Figure 4.24: RMSE when two highly correlated sources are well-separated and out of the grid, $L = 100$

From Figures 4.22 and 4.24, it is seen that the OGSBI method is not robust to the noise because there is too much increase in the RMSE when SNR is smaller than 10 dB. However, the method outperforms the others when SNR is more than 10 dB, for the sources are out of the grid. This shows the superiority of OGSBI over ℓ_1-SVD for off-grid sources.

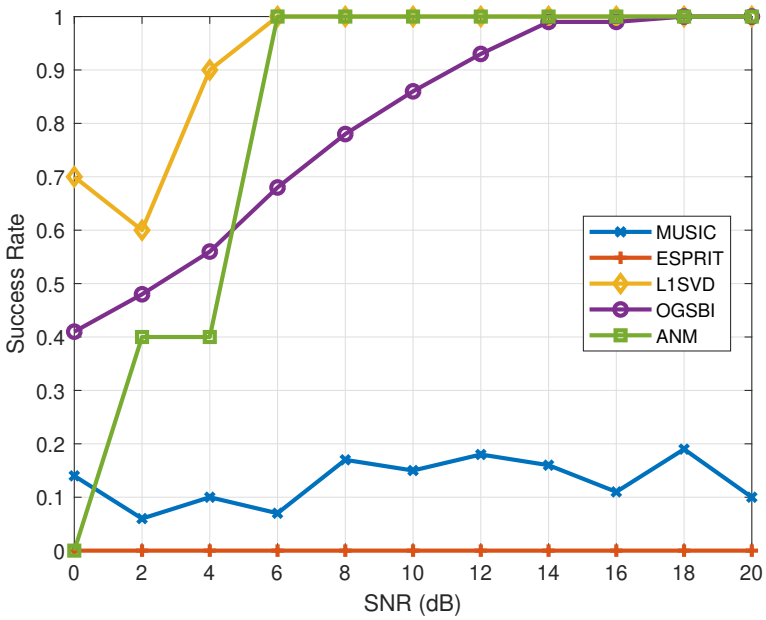

Figure 4.25: Success rate when two highly correlated sources are well-separated and out of the grid, $L = 100$

4.1.2.4 Scenario-IV: Effect of Discretization

In this scenario, we investigate the performance of grid-based methods with different grid intervals in the MMV parametric reconstruction problem. The grid interval is an essential parameter for the grid-based sparse methods, on-grid, and off-grid sparse methods. There is a trade-off between the estimation accuracy and the computational complexity due to the grid interval.

We consider 10 dB SNR and 0.5, 1 and 2° grid intervals. The sources are located at 10.5° and 40.5°. It is seen from the Table 4.3, ℓ_1-SVD reconstructs the arrival angles perfectly when the grid interval is 0.5°. However, if the grid interval is selected so that the sources being out of the grid, then ℓ_1-SVD estimates the angles that are on the closest grid point. On the other hand, OGSBI estimates the sources with small RMSE even if we select a large grid interval.

Table 4.3: RMSE vs. resolution when SNR is 20 dB

	RMSE		**Run Time**	
Grid Interval	**ℓ_1-SVD**	**OGSBI**	**ℓ_1-SVD**	**OGSBI**
0.5	0	0.1351	9.1099	6.4130
1	0.5	0.1442	4.6321	0.6978
2	0.5	0.4548	2.4780	0.1460

Run times of both methods are inversely proportional to the grid interval. It is expected because when the grid interval decreases, the number of columns in the dictionary matrix increases; consequently, the sparse approximation problem becomes larger. Also, it is seen that there is a jump between run times of 0.5° and 1° for both methods. The grid interval is selected as 1°, and this value is the most efficient in terms of time.

4.1.2.5 Scenario-V: Five Well-Separated Uncorrelated Sources

This scenario shows that the performance of the classical and sparsity-based methods when the number of sources is increased to 5. The direction-of-arrival with a large number of sources is encountered in practical applications [63]. The sources are well-separated because we would like to see only the effect of a large number of sources on the methods.

The sources are located at $-50°$, $-20°$, $10°$, $40°$ and $70°$.

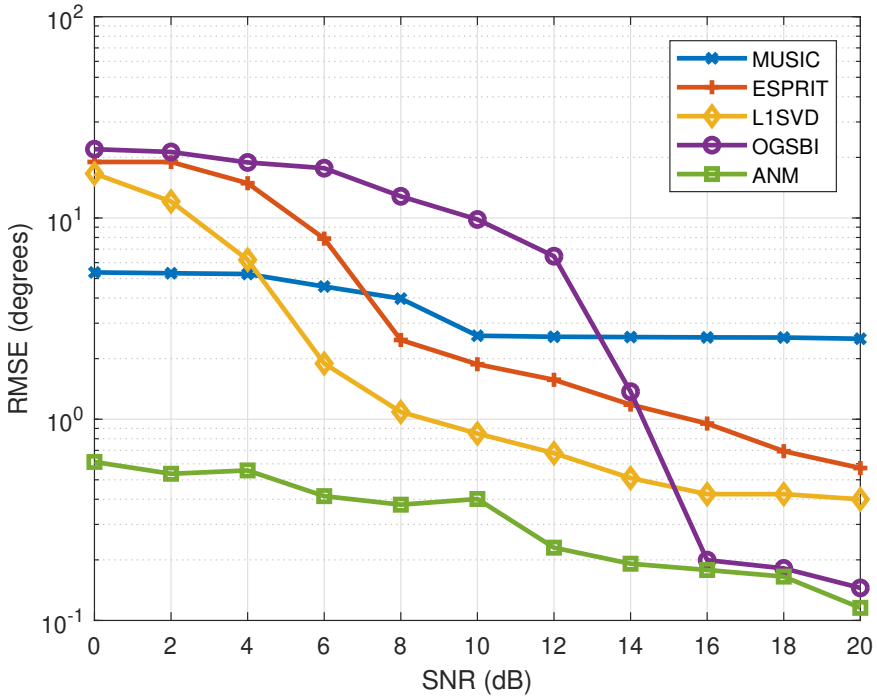

Figure 4.26: RMSE when five uncorrelated sources are well-separated and on the grid, $L = 100$

The performance of MUSIC is not satisfactory in the case of five sources presented in the problem. It is seen from Figure 4.27 that the success rate of the method is 0 in all SNR range. The other classical method, ESPRIT, also shows poor performance because its success rate is lower than that of sparsity-based methods in all SNR range.

It is seen from Figures 4.26 and 4.27 that ANM outperforms the other methods in all SNR range. However, the execution time of this method is too long. The solvers for

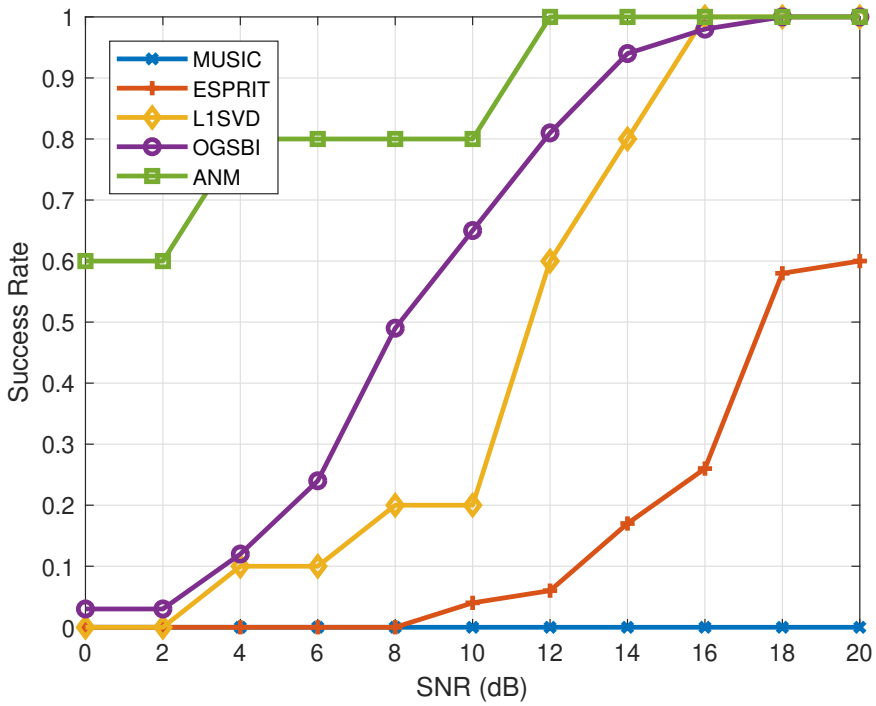

Figure 4.27: success rate when five uncorrelated sources are well-separated and on the grid, $L = 100$

the SDP formulation of the atomic norm are not computationally efficient, as said in Section 3.6.2.1.

Although the sources are on the grid, ℓ_1-SVD cannot reconstruct the arrival angles perfectly as it is seen from Figure 4.26. The reason is that there is a large number of sources that affect the sparse recovery algorithm in terms of sparsity.

Similar to the previous scenario, OGSBI is not robust to noise because there is a jump in the RMSE at 14 dB SNR in Figure 4.26. This weakness can be explained via the violation of the assumptions related to the noise in the OGSBI method. When the noise contribution decreases, the performance of the method is comparable to ANM when the SNR is more than 14 dB.

4.2 Parameterized Source Separation

The previous section was about the performance comparison of the classical and sparsity-based methods for one of the parametric reconstruction problems, DOA estimation. In addition to the comparison between sparsity-based and classical methods, the sparsity-based methods have been compared within themselves. The on-grid, off-grid, and gridless sparse methods were compared in terms of estimation accuracy and execution time. The methods that apply to the SMV and MMV models were evaluated separately.

This section focuses only on the sparsity-based methods and the MMV problem. Specifically, the representative sparsity-based methods that exploit the structure of standard sparsity, group sparsity, and joint sparsity are compared in this section. We evaluate the BPDN method, which exploits the standard sparsity, block OMP method, which exploits the group sparsity and ℓ_1-SVD method, which exploits the joint sparsity structures. BPDN and block OMP algorithms are applied to each of the L single measurement vector problems separately. ℓ_1-SVD method is applied to the MMV problem directly to exploits the redundancy between the measurements.

Recall that the matrix-vector formulation of the parametric model of parameterized source separation is given by

$$\boldsymbol{y} = \boldsymbol{A}(\boldsymbol{\Theta})\boldsymbol{s} + \boldsymbol{w} \tag{4.11}$$

where $\boldsymbol{\Theta}$ consists of the delay and width parameters to be estimated. Also, $\boldsymbol{s}$ is the amplitude vector to be estimated. Note that, each spectrum has its own matrix-vector formulation. However, if it is assumed that the values of parameters are constant over the spectra, we can write all matrix-vector formulations together. Then, the matrix formulation is given by

$$\boldsymbol{Y} = \boldsymbol{A}(\boldsymbol{\Theta})\boldsymbol{S} + \boldsymbol{W} \tag{4.12}$$

To apply the sparsity based methods to this problem, the sparsity based problem formulation of the above formulations are required.

Recall that the dictionary matrix for sparse representation using two parameters is

given by

$$\boldsymbol{A}([\bar{\boldsymbol{\alpha}}, \bar{\boldsymbol{\beta}}]^T) = [a([\bar{\alpha}_1, \bar{\beta}_1]^T), ..., a([\bar{\alpha}_{K_1}, \bar{\beta}_1]^T), ..., a([\bar{\alpha}_1, \bar{\beta}_{K_2}]^T), ..., a([\bar{\alpha}_{K_1}, \bar{\beta}_{K_2}]^T)] \tag{4.13}$$

On the other hand, if the parameter β_n is selected as a fixed continuous-valued variable, then the dictionary matrix for sparse representation becomes

$$\boldsymbol{A}([\bar{\boldsymbol{\alpha}}, \boldsymbol{\beta}]^T) = [a([\bar{\alpha}_1, \beta_1]^T), ..., a([\bar{\alpha}_{K_1}, \beta_1]^T), ..., a([\bar{\alpha}_1, \beta_N]^T), ..., a([\bar{\alpha}_{K_1}, \beta_N]^T)] \tag{4.14}$$

where β_N is the fixed continuous-valued parameter of the n$^{\text{th}}$ source signal.

In the parameterized source separation problem, the parameters α and β corresponds with delay and width parameters, respectively.

We will use two different sparsity-based forward problem formulation for this problem. In the first formulation, the sparsity is provided through both delay and width parameters. In the second formulation, the sparsity is provided through only delay parameter. The reason to use the second formulation is that the discretization of the width parameter leads to highly coherent dictionary which is difficult to solve using sparsity based methods. The first formulation using Equation (4.13) is given by

$$\boldsymbol{y} = \boldsymbol{A}([\bar{\boldsymbol{\alpha}}, \bar{\boldsymbol{\beta}}]^T)\boldsymbol{x} + \boldsymbol{w} \tag{4.15}$$

where $\boldsymbol{y}, \boldsymbol{w} \in \mathbb{R}^T$, $\boldsymbol{A}([\bar{\boldsymbol{\alpha}}, \bar{\boldsymbol{\beta}}]^T) \in \mathbb{R}^{T \times (K_1 K_2)}$ and $\boldsymbol{x} \in \mathbb{R}^{K_1 K_2}$. The second formulation uses Equation (4.14) as a dictionary matrix:

$$\boldsymbol{y} = \boldsymbol{A}([\bar{\boldsymbol{\alpha}}, \boldsymbol{\beta}]^T)\boldsymbol{x} + \boldsymbol{w} \tag{4.16}$$

where $\boldsymbol{\beta}$ is fixed continuous-valued width vector, $\boldsymbol{y}, \boldsymbol{w} \in \mathbb{R}^T$, $\boldsymbol{A}([\bar{\boldsymbol{\alpha}}, \boldsymbol{\beta}]^T) \in \mathbb{R}^{T \times (K_1 N)}$ and $\boldsymbol{x} \in \mathbb{R}^{K_1 N}$.

The noise samples are selected from i.i.d. Gaussian distribution $w_i \sim \mathcal{N}(0, \sigma^2)$, $i = 1, 2, ..., T$. SNR is defined as ten times of the $\log$-ratio of the power of the noiseless signal and the noise variance, σ^2. For Equation (4.15), SNR is given by

$$\text{SNR} = 10 \log_{10} \frac{\|\boldsymbol{A}([\bar{\boldsymbol{\alpha}}, \bar{\boldsymbol{\beta}}]^T)\boldsymbol{x}\|_2^2 / T}{\sigma^2} \tag{4.17}$$

For Equation (4.16), SNR is given by

$$\text{SNR} = 10\log_{10}\frac{\|\boldsymbol{A}([\bar{\boldsymbol{\alpha}},\boldsymbol{\beta}]^T)\boldsymbol{x}\|_2^2/T}{\sigma^2} \tag{4.18}$$

The estimation of true parameters in Equation (4.15) can be obtained directly, using sparsity-based methods. However, the dictionary matrix in the linear sparse model in Equation (4.15) is highly coherent due to the discretization of the width parameter. We expect that the sparsity-based methods will not able to obtain an accurate result. On the other hand, to obtain the true parameters in Equation (4.16), we have to estimate the width parameter using a continuous-valued parameter estimation method. We use the alternating least-squares algorithm to solve the parameterized source separation problem with the model in Equation (4.16). In the first stage of this algorithm, the width parameter is estimated using the Levenberg-Marquardt algorithm, which solves the nonlinear least-squares problem. In the second-stage, amplitude and delay parameters are estimated using the sparsity-based methods, as mentioned earlier.

The experiment setting that is common to all scenarios is given in Table 4.4.

Table 4.4: Common experiment settings for parameterized source separation

SNR	Varies (given in Eq. (4.17) and (4.18))
Wavelength samples(T)	150
Delay range	[0,150]
Delay discretization step(K_1)	0.5
Width range	[1,6]
Width discretization step(K_2)	0.5

We use amplitude, delay, width error and run time as performance metrics to measure the reconstruction accuracy and speed of the methods on the parameterized source separation problem. The errors for the parameters are defined as follows:

$$\sum_l \frac{\|\hat{\boldsymbol{c}}_{\boldsymbol{l}} - \boldsymbol{c}_{\boldsymbol{l}}^*\|_2^2}{\|\boldsymbol{c}_{\boldsymbol{l}}^*\|_2^2} \tag{4.19}$$

$$\sum_l \frac{\|\hat{\boldsymbol{a}}_{\boldsymbol{l}} - \boldsymbol{a}_{\boldsymbol{l}}^*\|_2^2}{\|\boldsymbol{a}_{\boldsymbol{l}}^*\|_2^2} \tag{4.20}$$

$$\sum_{l} \frac{\|\hat{\boldsymbol{w}}_l - \boldsymbol{w}_l^*\|_2^2}{\|\boldsymbol{w}_l^*\|_2^2} \tag{4.21}$$

where the index l denotes the l$^{\text{th}}$ spectrum and $\boldsymbol{c}_l^*$, $\boldsymbol{w}_l^*$ and $\boldsymbol{a}_l^*$ are ground-truth parameters.

4.2.1 Effect of Discretization

In this section, the aim is to see the effect of the discretization over the width parameter that leads to a highly coherent dictionary. To do this, we obtain a spectrum consists of two Gaussian sources that have width and delay parameters on the grid points. The parameters of the Gaussian sources is given in Table 4.5. We set the SNR to 15 dB.

Table 4.5: Ground-truth parameter values for Gaussian sources

	First Source	**Second Source**
Width(w)	2	4
Delay(c)	60	120
Amplitude(s)	5.22	3.36

We use the sparse representations in Equation (4.15) and (4.16) for the above parametric model. The basis pursuit denoising algorithm is selected as a representative for the sparsity-based methods because the parameters in the model are on the grid, and BPDN is a fast on-grid sparse method that uses the ADMM scheme. The parameters in the linear sparse model in Equation (4.15) directly estimated using the BPDN algorithm. On the other hand, the parameters in the model in Equation (4.16) are estimated using the ALS algorithm. Amplitude and delay parameters are estimated using the Levenberg-Marquardt algorithm, which solves a nonlinear least-squares problem in the first stage of the alternating algorithm. In the second stage, the width parameter is estimated using BPDN.

Figure 4.28 shows the spectrum with two Gaussian sources using 100 Monte Carlo simulations. It is easily seen from Table 4.6 that the ALS algorithm outperforms the

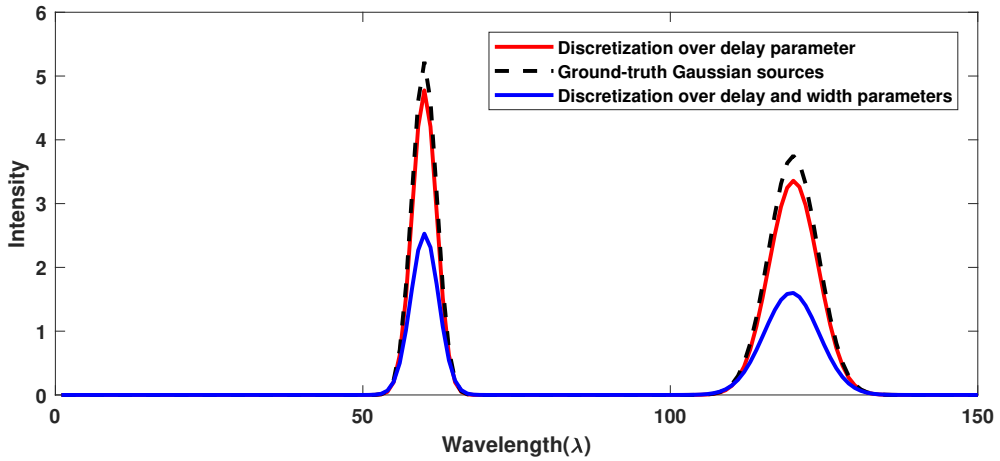

Figure 4.28: Ground-truth and estimated Gaussian sources as a result of 100 Monte Carlo simulations

BPDN algorithm. It is expected because of the parametric models. BPDN uses the linear sparse model that has a highly coherent dictionary. The reason for the excessive estimation error in the amplitude parameter is that BPDN estimates a sparse vector that has nonzero values corresponds to the wrong dictionary atoms. These wrong atoms are highly coherent with true atoms.

Table 4.6: Estimation errors for parameters

	ALS	**BPDN**
Width error	**0.0037**	0.2236
Delay error	**0.0014**	0.0035
Amplitude error	**0.0911**	0.5201

In the following section, we decide to use the sparse representation in Equation (4.16) because the parametric reconstruction based on the other representation is not reasonable.

4.2.2 Comparison of Different Methods

The estimation results in the previous section indicate that the direct application of sparsity-based methods to parameterized source separation problem shows poor per-

formance because of the highly coherent dictionary. In this section, we proceed with the model in Equation (4.16).

Recall that we cannot use the sparsity-based algorithms directly to estimate width, delay, and amplitude parameters since the width parameter appears in the model as a fixed continuous-valued variable. We use the alternating least-squares algorithm as a primary method. The sparsity-based methods take place in the amplitude and delay estimation stage of the algorithm. The width parameter is estimated in the other stage using Levenberg-Marquardt, a nonlinear least-squares algorithm.

We select the sources on the grid points for all scenarios in this section because the goal of the comparison is to compare the performance of the methods that consider the standard sparsity, group sparsity, and joint sparsity. All sparsity-based methods that used in this experiment are on-grid methods but consider different sparsity structures.

4.2.2.1 Scenario-I: Four Well-Separated Sources

This scenario is most straightforward, and it provides a baseline for the performance of the methods. The number of spectrum is selected as $L = 30$. The delay parameter values are selected to be equispaced with $\boldsymbol{c}_l^* = [20, 60, 100, 140]^T$ for all l. The width parameters are also selected to be equispaced with $\boldsymbol{w}_l^* = [1, 2.5, 4, 5.5]^T$ for all l. Note that the delays of Gaussian sources can be slowly changed between the spectra in practical applications, but we consider them as constant in this basic configuration to see the performance of the method that enforces joint sparsity. Figure 4.29 shows the sample ground-truth Gaussian sources when SNR is 15 dB for all spectra.

The Figures 4.30, 4.32 and 4.31 show the delay, width and amplitude errors of sparsity-based methods, respectively. The average run times of the methods are in the Table 4.7.

The sparse representation of this experiment setting has the joint sparse structure because the columns of the ground-truth sparse matrix have common support. Also, each column vector has the group sparse structure because each sub-vector of the column vector that corresponds to different width values is 1-sparse. ℓ_1-SVD uses $\ell_{2,1}$-norm in the inverse problem formulation to exploit the joint sparsity but the other

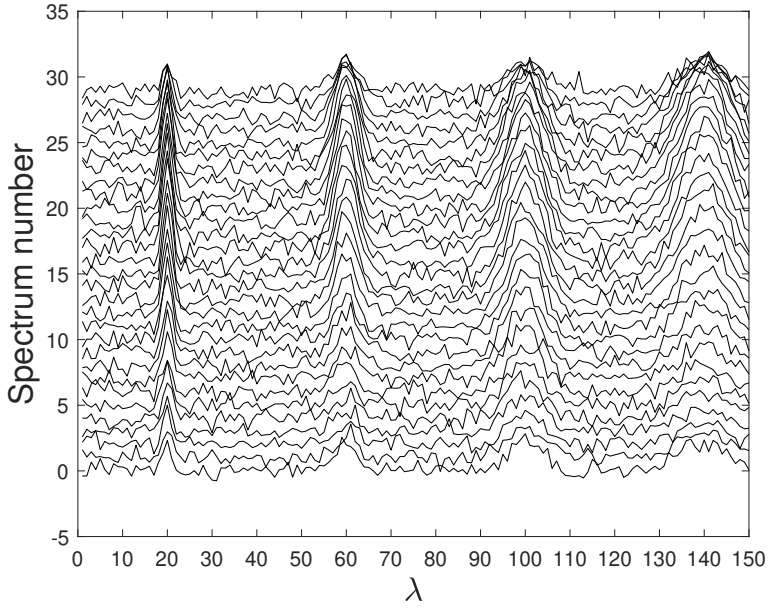

Figure 4.29: Sample measurement for 4 well separated sources with 15 dB SNR

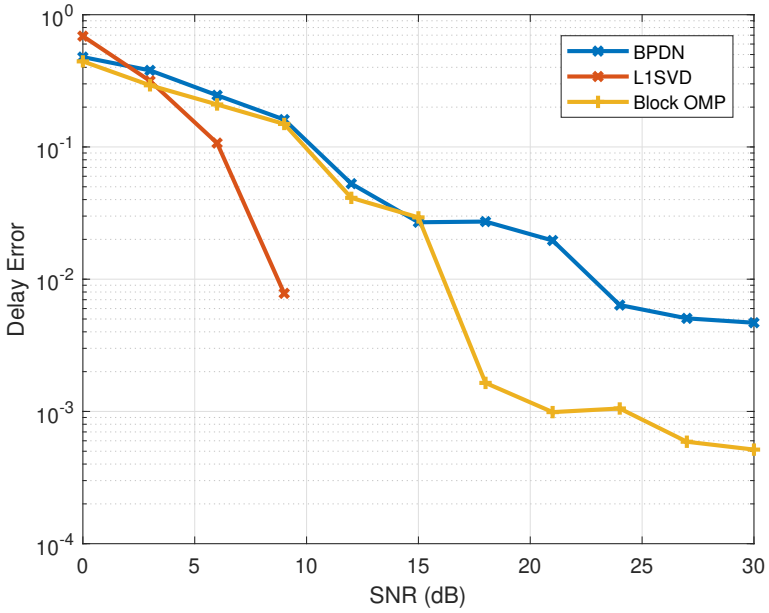

Figure 4.30: Delay error for 4 well separated sources

methods not. As a result, it is seen from Figure 4.30 that ℓ_1-SVD outperforms the other methods almost all SNR range. The block OMP exploits the group sparsity in each of the column vectors using a block ℓ_1-norm, but it is not successful as much as

ℓ_1-SVD. This shows us that the joint sparsity prior is more dominant than the group sparsity prior to each column vector. We know that BPDN does not consider any type of structured sparsity; it only considers the standard sparsity. Then, it exploits neither group sparsity in the column vector nor the joint sparsity between the column vectors. Consequently, the other methods exploit structured sparsity outperform BPDN.

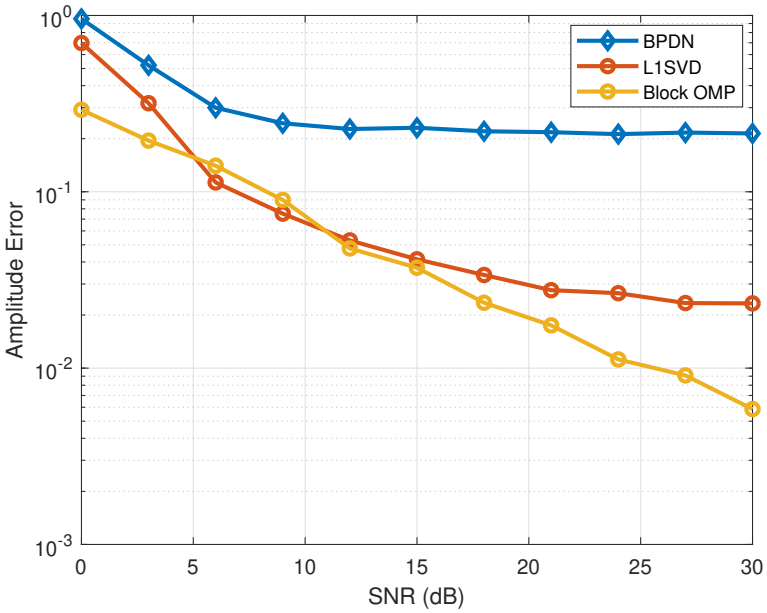

Figure 4.31: Amplitude error for 4 well separated sources

BPDN is not as successful as ℓ_1-SVD and block OMP for the estimation of the amplitude parameter. The reason is that it distributes the amplitude estimates to the sources that similar to the true ones. Block OMP is designed so that it estimates only 4 nonzero entries in the sparse vector that corresponds to each spectrum. Similarly, $\ell_{2,1}$-norm in the inverse problem of ℓ_1-SVD forces to estimate the same number of sources in each spectrum. Thus, the amplitude estimates are not distributed in these methods.

All methods estimate the width parameters with almost the same error, as it is seen in Figure 4.32. It is expected because we have used the same algorithm in the width estimation stage for all methods.

Block OMP is an instance of greedy methods. The greedy methods are the fastest among the sparsity-based methods. Table 4.7 follows this fact. BPDN is faster than

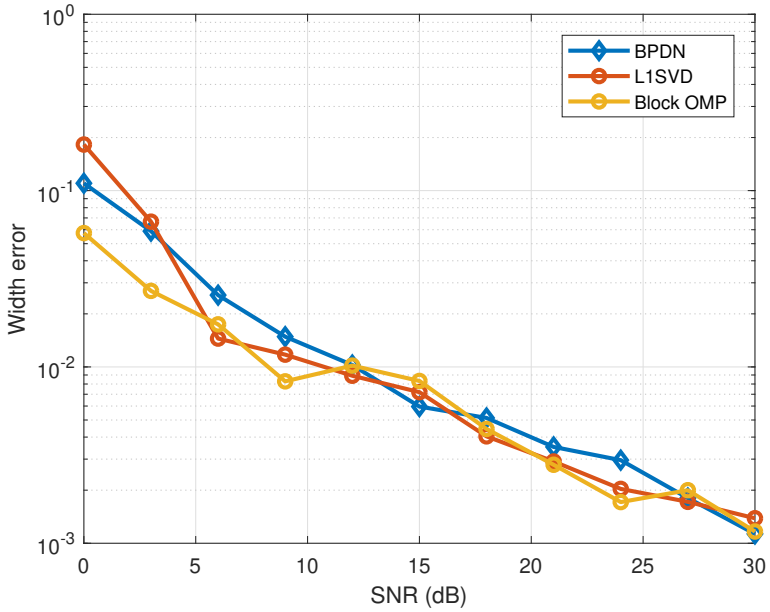

Figure 4.32: Width error for 4 well separated sources

Table 4.7: Average run times of methods for Scenario-I

Method	**Average Run time (s)**
BPDN	8.0473
L1-SVD	12.1561
Block OMP	0.5979

ℓ_1-SVD because it uses ADMM rather than SOC programming. Note that we will not include the run times for the other scenarios because the run times of the methods are similar for each scenario.

4.2.2.2 Scenario-II: Four Closely-Spaced Sources

We would like to see the effect of being closely-spaced sources on the estimation accuracy in this scenario. To do this, the delay parameter values are selected as $\boldsymbol{c}_l^* = [20, 40, 120, 140]^T$ for all l. The width parameters are selected to be equispaced with $\boldsymbol{w}_l^* = [1, 2.5, 4, 5.5]^T$ for all l being same as the previous scenario. We keep the width parameters constant because estimation of width parameters are not focus of

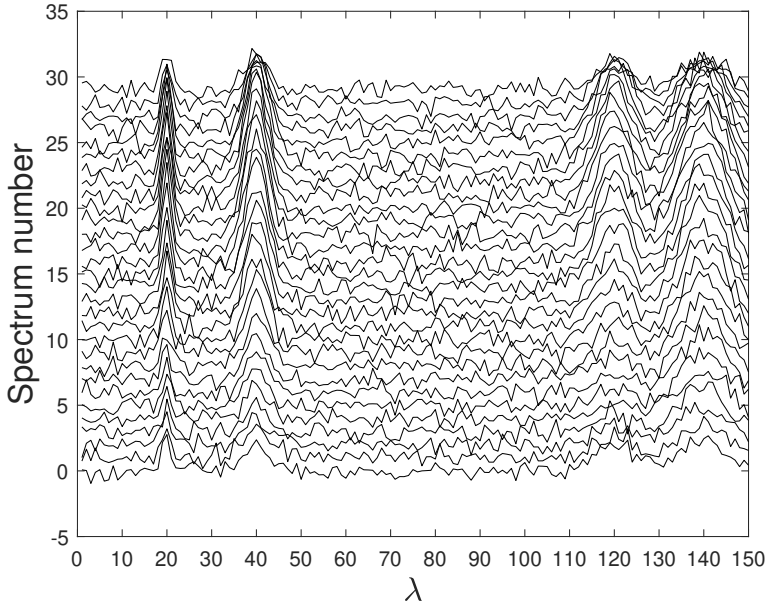

Figure 4.33: Sample measurement for 4 closely-spaced sources with 15 dB SNR

this experiment. The number of spectrum is selected as $L = 30$. Figure 4.33 shows the sample ground-truth Gaussian sources when SNR is 15 dB for all spectra.

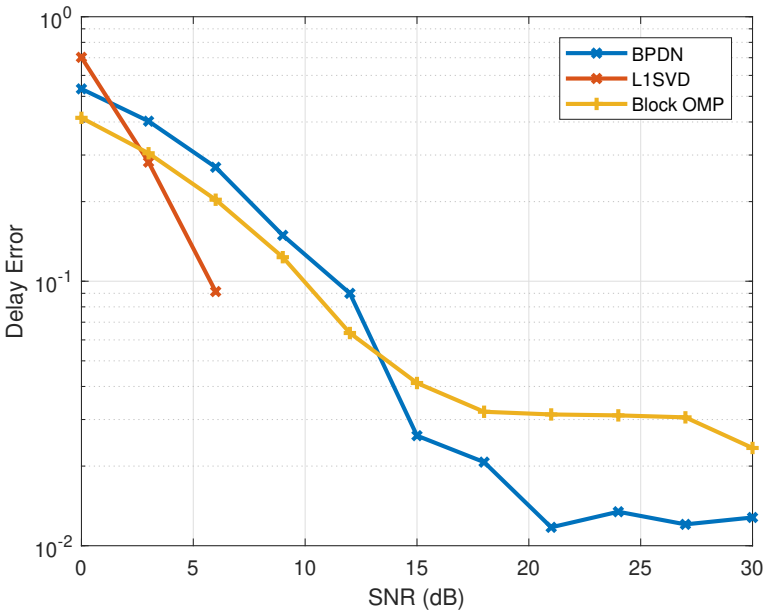

Figure 4.34: Delay error for 4 closely-spaced sources

The Figures 4.34, 4.35 and 4.36 show the delay, amplitude and width errors of sparsity-

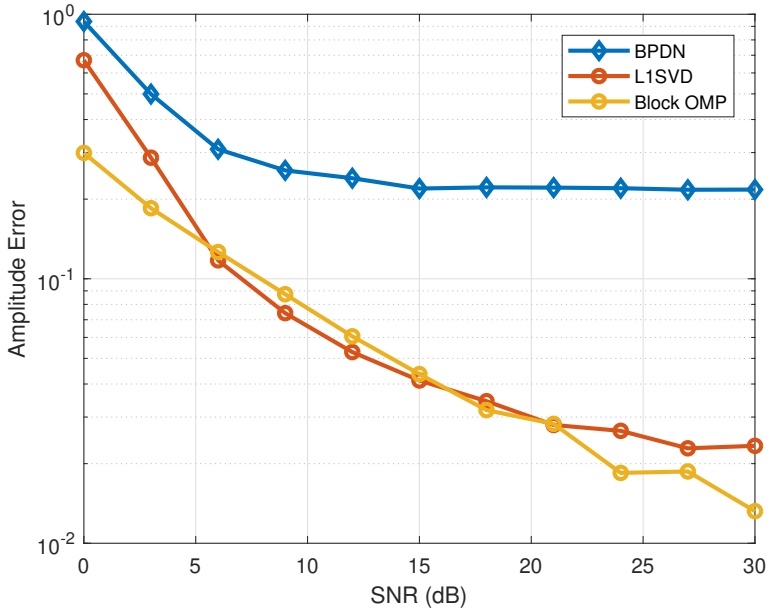

Figure 4.35: Amplitude error for 4 closely-spaced sources

based methods, respectively.

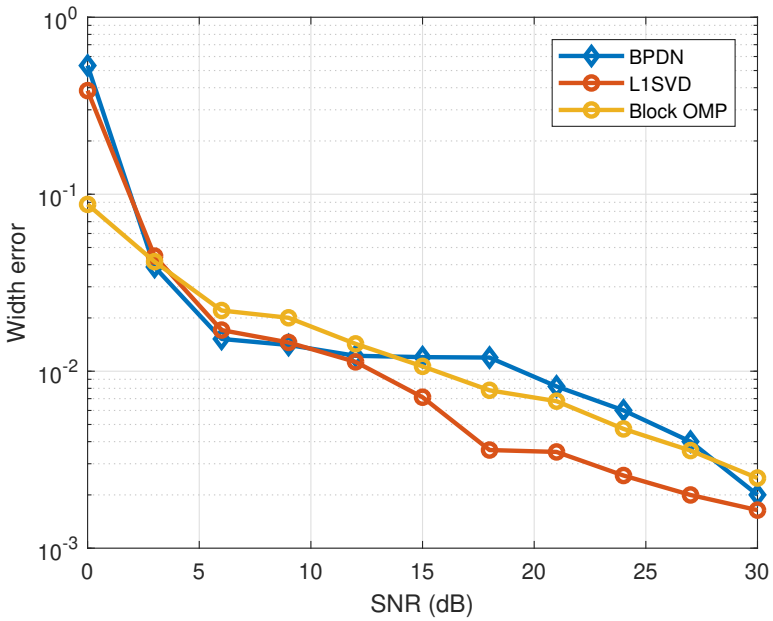

Figure 4.36: Width error for 4 closely-spaced sources

We see that ℓ_1-SVD outperforms the other methods in the delay estimation as in Scenario-I from Figure 4.34. Also, BPDN shows a similar characteristic to the Scenario-

I. However, block OMP cannot maintain its performance when the sources are closely-spaced.

Figures 4.36 and 4.35 show almost the same results with that of Scenario-I. It is expected because the width estimation algorithm is the same, and the BPDN method distributes the amplitude estimates to the other entries in the sparse column vectors, unlike the other methods.

4.2.2.3 Scenario-III: Four Well Separated Sources with Slow Evolution of Delay Parameter

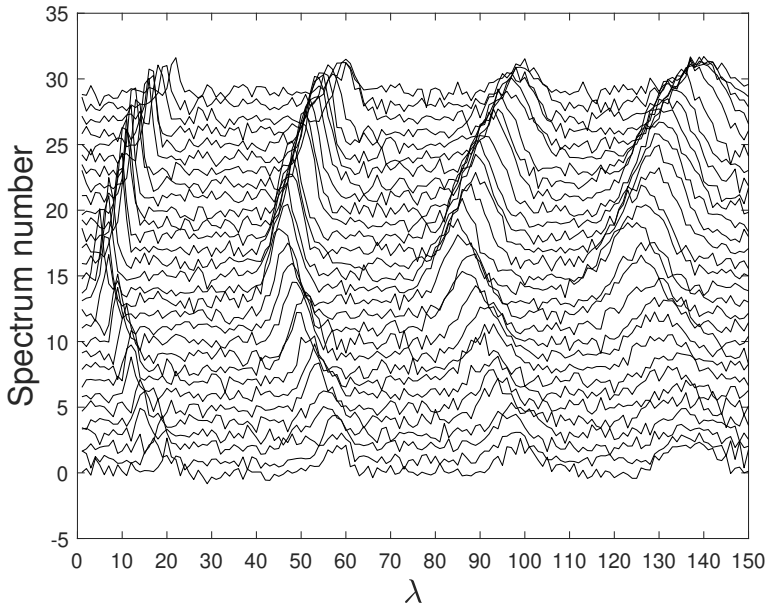

Figure 4.37: Sample measurement for 4 well separated sources with slow evolution of delay parameter with 15 dB SNR

In this scenario, we select the delay parameters so that the values change slowly between the spectra. This scenario is encountered in practical applications such as galaxy kinematics, where each spectrum represents the different pixels in the astrophysical image. The delay difference between the sources is set to 40 to separate the sources well. The width parameters are again selected to be equispaced with $\boldsymbol{w}_l^* = [1, 2.5, 4, 5.5]^T$ for all l. Figure 4.37 shows the sample ground-truth Gaussian sources when SNR is 15 dB for all spectra.

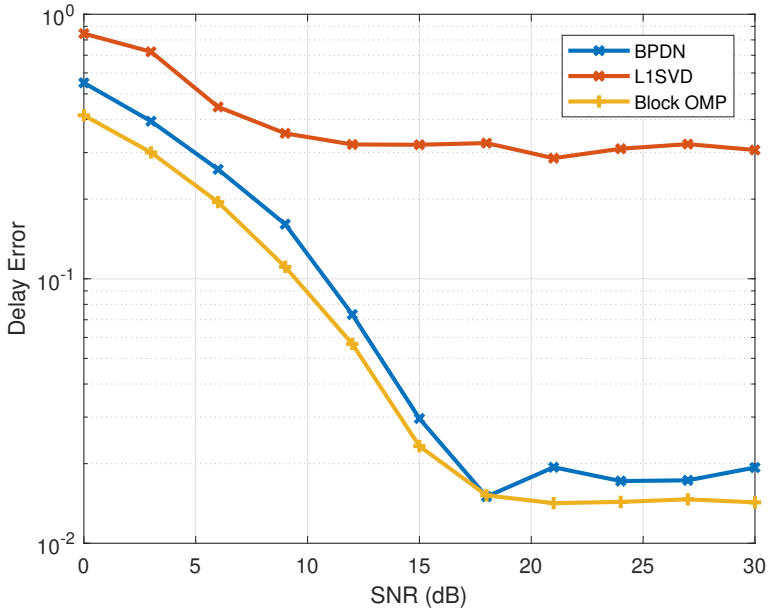

Figure 4.38: Delay error for 4 well separated sources with slow evolution of delay parameter

ℓ_1-SVD shows poor performance in this scenario as expected as it is seen in Figure 4.38, 4.39 and 4.40. This scenario does not provide joint sparsity between the sparse column vectors because the delay parameters change between the spectra. However, ℓ_1-SVD tries to estimate joint sparse vectors using $\ell_{2,1}$-norm. This causes to high estimation error of delay and width parameters, consequently the amplitude parameter.

If we compare 4.38 and 4.30, we see that BPDN shows nearly the same performance as in Scenario-I because the value of delay parameters is not essential for BPDN if the separation between them is the same. On the other hand, the delay parameter estimate of block OMP is not accurate as much as in Scenario-I. Block OMP is a heuristic algorithm like standard OMP. Thus, it is not robust to the change in the value of delay parameters even if the separation between them is the same.

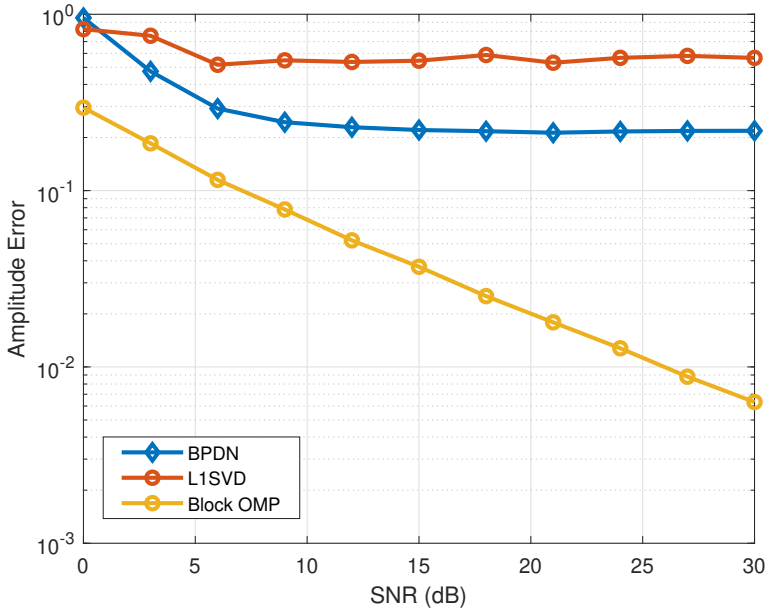

Figure 4.39: Amplitude error for 4 well separated sources with slow evolution of delay parameter

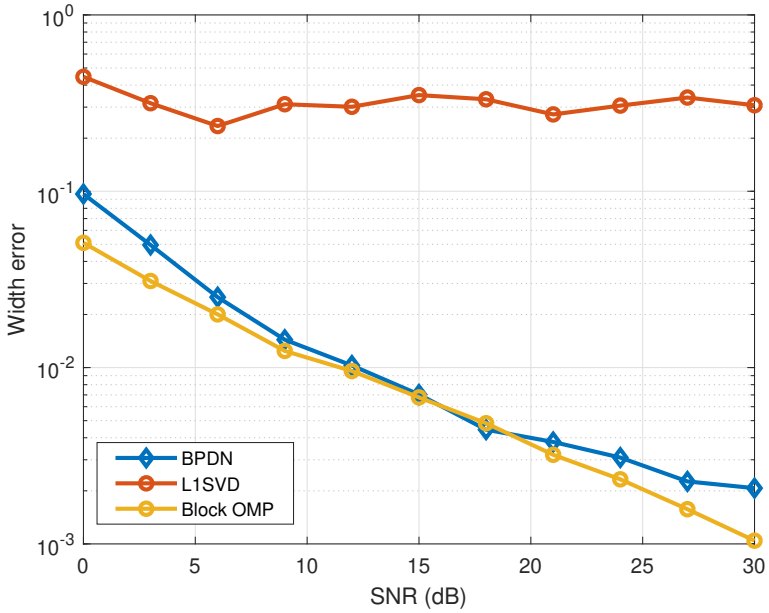

Figure 4.40: Width error for 4 well separated sources with slow evolution of delay parameter

CHAPTER 5

CONCLUSIONS

In this thesis, we have studied the problem of parametric recovery and evaluated the various solution approaches for this problem. In a parametric recovery problem, the unknown model parameters in each superimposed signal are estimated from noisy observations. The goal is to estimate the unknown continuous-valued parameters of the functions from the noisy data. The solution approaches are divided into two categories, namely, classical and sparsity-based methods. Classical methods directly work on the continuous-valued parameter space and solve the resulting nonlinear parameter estimation problem, but the sparsity-based methods convert the nonlinear parametric recovery problem into a linear sparse recovery problem of larger dimension. The conventional approach in the sparsity-based methods is to discretize the parameter space to form a dictionary whose atoms correspond to candidate parameter values, represent the data as a linear combination of a small number of dictionary atoms, and then solve the resulting linear inverse problem.

We have presented the mathematical model for the forward problem for both classical and sparsity-based methods. We have obtained mathematical models for single measurement vector and multiple measurement vector cases, separately, since these cases have different difficulty level. The mathematical model for the classical forward problem was straightforward, but it is not trivial for the sparsity-based forward problem. We have obtained a dictionary by discretizing the continuous-valued parameter set. However, the number of atoms in the dictionary is equal to the multiplication of the number of elements in the discrete sets of parameter values if the number of parameters is more than one. In this case, the number of atoms increases exponentially, and the mutual coherence of the dictionary increases. As a result, the linear inverse

problem becomes larger and hard to solve.

Classical methods generally uses the maximum likelihood framework. Among them, we evaluate the nonlinear least-squares algorithm and the different subspace-based methods, MUSIC, and ESPRIT. Subspace-based methods could achieve performance near to maximum likelihood in many cases. However, these methods suffer from both the source correlation and fewer number of measurements.

Sparsity-based methods are developed for the parametric recovery problem in the last decades to mitigate the issues that are encountered in classical methods. These methods are part of the regularized least-squares method with different sparsity regularizers. We have reviewed these methods in three categories, on-grid, off-grid, and gridless sparse methods. On-grid methods require that the true parameter values lie on a set of fixed grid points. Thus, it is inevitable to estimate the parameter values only in the fixed grid points. We have evaluated OMP, block OMP, BPDN, FOCUSS, ℓ_1-SVD, and ALS algorithms as on-grid methods. Off-grid methods also use a grid, but the recovered parameter values are allowed to be out of the grid points. The methods of SURE-IR and OGSBI have been evaluated in this category. Gridless sparse methods do not require a grid, and they work directly in the continuous-valued parameter space. We have evaluated only the ANM method, which is the most known in this category. The other gridless sparse methods are not presented in this thesis because these methods are introduced very recently in the literature and do not have substantial theoretical backgrounds.

We have compared the different methods in two parametric recovery problems, direction-of-arrival, and parameterized source separation. Direction-of-arrival has one parameter to be estimated, the arriving angle. On the other hand, the parameterized source separation has two parameters to be estimated, delay, and width parameters.

The dictionary has been obtained using the discrete set of arriving angle values for direction-of-arrival. Then, the classical and sparsity-based methods have been compared in terms of estimation accuracy and computational complexity for both single snapshot and multiple snapshot direction-of-arrival. Especially, FOCUSS, BPDN, SURE-IR, OGSBI, and ANM methods have been compared in a single snapshot problem. The classical methods were not in this comparison because they require

a sufficient number of the snapshot. In the case of multiple snapshots, MUSIC, ESPRIT, BPDN, OGSBI, and ANM have been compared. For both cases, we have seen that the ANM method outperforms the others in most cases, but the execution time of it substantially higher than others. On the other hand, we have noticed that the off-grid methods, SURE-IR and OGSBI, show poor performance in the single snapshot DOA estimation problem. The reason is that they require a certain number of measurement vectors to show good performance since they use several numbers of hyper-parameters. Lastly, the on-grid sparse methods, FOCUSS, and BPDN show moderated performance in terms of both estimation accuracy and speed.

In parameterized source separation application, there are two parameters, delay, and width of Gaussian sources. We have obtained two different dictionaries for the parameterized source separation, one by discretizing the delay parameter only and the other by discretizing both parameters. The former dictionary is obtained so that it has lower mutual coherence than the latter dictionary. The width parameter in the former dictionary is estimated a priori and does not take part in sparse recovery. In the first experiment, we have compared the results using these dictionaries and concluded that the dictionary discretizing both parameters causes poor estimation performance. We have decided to use the dictionary discretizing the delay parameter only. As a primary method, we have used a two-stage ALS algorithm. In the first stage, the width parameter is estimated using a nonlinear least-squares algorithm, and in the second stage, it estimates delay parameter using any sparsity-based methods. We have compared BPDN, block OMP, and ℓ_1-SVD methods as the second stage of the ALS algorithm for this problem. We have concluded that ℓ_1-SVD outperforms the other methods if the values of delay parameters do not change over different measurements, and block OMP outperforms the others if the delay parameters are not constant over different measurements in terms of estimation accuracy. On the other hand, block OMP outperforms the other methods for all scenarios in terms of speed because it is a greedy method.

Although block OMP outperforms the other methods when the delay parameters are not constant over different measurements, it does not fully leverage the structure of the sparse matrix. The reason is that the delay values change slowly in the real scenarios, but the block OMP treats the MMV problem as a certain number of SMV

problems. Thus, it cannot exploits the redundancy between the measurements. As future work, we would like to obtain an inverse problem formulation that leverages the slow change in the delay parameter and a solution approach to this inverse problem.

REFERENCES

[1] P. Stoica and K. C. Sharman, "Maximum likelihood methods for direction-of-arrival estimation," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 38, pp. 1132–1143, July 1990.

[2] M. Lustig, D. L. Donoho, J. M. Santos, and J. M. Pauly, "Compressed sensing MRI," *IEEE Signal Processing Magazine*, vol. 25, pp. 72–82, March 2008.

[3] P. Stoica and R. Moses, "Spectral analysis of signals," *Prentice Hall*, 01 2005.

[4] L. C. Potter and R. L. Moses, "Attributed scattering centers for SAR ATR," *IEEE Transactions on Image Processing*, vol. 6, pp. 79–91, Jan 1997.

[5] F. S. Oktem, F. Kamalabadi, and J. M. Davila, "A parametric estimation approach to instantaneous spectral imaging," *IEEE Transactions on Image Processing*, vol. 23, pp. 5707–5721, Dec 2014.

[6] S. Minin and F. Kamalabadi, "Uncertainties in extracted parameters of a gaussian emission line profile with continuum background," *Appl. Opt.*, vol. 48, pp. 6913–6922, Dec 2009.

[7] J. Ireland, "Precision limits to emission-line profile measuring experiments," *The Astrophysical Journal*, vol. 620, pp. 1132–1139, feb 2005.

[8] J. M. Davila, F. S. Oktem, and F. Kamalabadi, "Slitless solar imaging spectroscopy," *The Astrophysical Journal*, vol. 883, p. 7, sep 2019.

[9] H. Mortada, V. Mazet, C. Soussen, C. Collet, and L. Poisson, "Parameterized source separation for delayed spectroscopic signals," *Signal Processing*, vol. 158, pp. 48 – 60, 2019.

[10] S. Som, L. C. Potter, R. Ahmad, D. S. Vikram, and P. Kuppusamy, "Epr oximetry in three spatial dimensions using sparse spin distribution," *Journal of Magnetic Resonance*, vol. 193, no. 2, pp. 210 – 217, 2008.

[11] D. Luengo, J. Vía, S. Monzón, T. Trigano, and A. Artés-Rodríguez, "Cross-products LASSO," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 6118–6122, May 2013.

[12] Y. Sepulcre, T. Trigano, and Y. Ritov, "Sparse regression algorithm for activity estimation in γ spectrometry," *IEEE Transactions on Signal Processing*, vol. 61, pp. 4347–4359, Sep. 2013.

[13] J. Fang, F. Wang, Y. Shen, H. Li, and R. S. Blum, "Super-resolution compressed sensing for line spectral estimation: An iterative re-weighted approach," *IEEE Transactions on Signal Processing*, vol. 64, pp. 4649–4662, Sep. 2016.

[14] N. Moal and J. . Fuchs, "Sinusoids in white noise: A quadratic programming approach," in *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP '98 (Cat. No.98CH36181)*, vol. 4, pp. 2221–2224 vol.4, May 1998.

[15] P.-J. Chung, M. Viberg, and J. Yu, "Chapter 14 - DOA estimation methods and algorithms," in *Academic Press Library in Signal Processing: Volume 3* (A. M. Zoubir, M. Viberg, R. Chellappa, and S. Theodoridis, eds.), vol. 3 of *Academic Press Library in Signal Processing*, pp. 599 – 650, Elsevier, 2014.

[16] D. Malioutov, "A sparse signal reconstruction perspective for source localization with sensor arrays," Master's thesis, Massachusetts Institute of Technology, Massachusetts, 7 2003.

[17] E. Villeneuve and H. Carfantan, "Nonlinear deconvolution of hyperspectral data with MCMC for studying the kinematics of galaxies," *IEEE Transactions on Image Processing*, vol. 23, pp. 4322–4335, Oct 2014.

[18] H. Krim and M. Viberg, "Two decades of array signal processing research: the parametric approach," *IEEE Signal Processing Magazine*, vol. 13, pp. 67–94, July 1996.

[19] G. Golub and V. Pereyra, "Separable nonlinear least squares: the variable projection method and its applications," *Inverse Problems*, vol. 19, pp. R1–R26, feb 2003.

[20] P. Deuflhard, *Least Squares Problems: Gauss-Newton Methods*, pp. 173–231. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011.

[21] R. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Transactions on Antennas and Propagation*, vol. 34, pp. 276–280, March 1986.

[22] K. Agarwal, L. Pan, Y. K. Leong, M. Han, O. Y. Chan, X. Chen, and S. P. Yeo, "Practical applications of multiple signal classification," *International Journal of RF and Microwave Computer-Aided Engineering*, vol. 22, no. 3, pp. 359–369, 2012.

[23] B. Friedlander, "Chapter 1 - wireless direction-finding fundamentals," in *Classical and Modern Direction-of-Arrival Estimation* (T. E. Tuncer and B. Friedlander, eds.), pp. 1 – 51, Boston: Academic Press, 2009.

[24] R. Roy and T. Kailath, "ESPRIT-estimation of signal parameters via rotational invariance techniques," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, pp. 984–995, July 1989.

[25] B. Subudhi, S. K. Sarnal, and S. Ghosh, "A new low-frequency oscillatory modes estimation using TLS-ESPRIT and least mean squares sign-data (LMSSD) adaptive filtering," in *TENCON 2017 - 2017 IEEE Region 10 Conference*, pp. 751–756, Nov 2017.

[26] D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, pp. 1289–1306, April 2006.

[27] J. A. Tropp and S. J. Wright, "Computational methods for sparse solution of linear inverse problems," *Proceedings of the IEEE*, vol. 98, pp. 948–958, June 2010.

[28] S. G. Mallat and Zhifeng Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on Signal Processing*, vol. 41, pp. 3397–3415, Dec 1993.

[29] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition," in *Proceedings of 27th Asilomar Conference on Signals, Systems and Computers*, pp. 40–44 vol.1, Nov 1993.

[30] D. Needell and J. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301 – 321, 2009.

[31] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, vol. 20, no. 1, pp. 33–61, 1998.

[32] I. Gorodnitsky and B. Rao, "Sparse signal reconstruction from limited data using FOCUSS: A re-weighted minimum norm algorithm," *IEEE Transactions on Signal Processing*, vol. 45, pp. 600–616, Mar. 1997.

[33] E. J. Candès, M. B. Wakin, and S. P. Boyd, "Enhancing sparsity by reweighted ℓ_1 minimization," *Journal of Fourier Analysis and Applications*, vol. 14, pp. 877–905, Dec 2008.

[34] S. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky, "An interior-point method for large-scale ℓ_1-regularized least squares," *IEEE Journal of Selected Topics in Signal Processing*, vol. 1, pp. 606–617, Dec 2007.

[35] J. Zou, H. Li, and G. Liu, "Split Bregman algorithm for structured sparse reconstruction," *IEEE Access*, vol. 6, pp. 21560–21569, 2018.

[36] Y. C. Eldar, P. Kuppinger, and H. Bolcskei, "Block-sparse signals: Uncertainty relations and efficient recovery," *IEEE Transactions on Signal Processing*, vol. 58, pp. 3042–3054, June 2010.

[37] E. J. Candès, J. K. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Communications on Pure and Applied Mathematics*, vol. 59, no. 8, pp. 1207–1223, 2006.

[38] A. Cohen, W. Dahmen, and R. DeVore, "Compressed sensing and best k-term approximation," *Journal of the American Mathematical Society*, vol. 22, no. 1, pp. 211–231, 2009.

[39] L. C. Potter, E. Ertin, J. T. Parker, and M. Cetin, "Sparsity and compressed sensing in radar imaging," *Proceedings of the IEEE*, vol. 98, pp. 1006–1020, June 2010.

[40] Y. Chi, L. L. Scharf, A. Pezeshki, and A. R. Calderbank, "Sensitivity to basis mismatch in compressed sensing," *IEEE Transactions on Signal Processing*, vol. 59, pp. 2182–2195, May 2011.

[41] M. F. Duarte and Y. C. Eldar, "Structured compressed sensing: From theory to applications," *IEEE Transactions on Signal Processing*, vol. 59, pp. 4053–4085, Sep. 2011.

[42] R. T. Rockafellar, *Convex analysis*, vol. 28. Princeton university press, 1970.

[43] S. J. Wright, R. D. Nowak, and M. A. T. Figueiredo, "Sparse reconstruction by separable approximation," *IEEE Transactions on Signal Processing*, vol. 57, pp. 2479–2493, July 2009.

[44] J. M. Bioucas-Dias and M. A. T. Figueiredo, "Alternating direction algorithms for constrained sparse regression: Application to hyperspectral unmixing," in *2010 2nd Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing*, pp. 1–4, June 2010.

[45] J. Eckstein and D. P. Bertsekas, "On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators," *Mathematical Programming*, vol. 55, pp. 293–318, Apr 1992.

[46] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, p. 1–122, Jan. 2011.

[47] N. Parikh and S. Boyd, "Proximal algorithms," *Found. Trends Optim.*, vol. 1, p. 127–239, Jan. 2014.

[48] M. Z. Nashed, "The theory of Tikhonov regularization for Fredholm equations of the first kind (C. W. Groetsch)," *SIAM Review*, vol. 28, no. 1, pp. 116–118, 1986.

[49] S. Boyd and L. Vandenberghe, *Convex Optimization*. USA: Cambridge University Press, 2004.

[50] J. F. Sturm, "Using SeDuMi 1.02, a matlab toolbox for optimization over symmetric cones," *Optimization Methods and Software*, vol. 11, pp. 625–653, jan 1999.

[51] M. S. Lobo, L. Vandenberghe, S. Boyd, and H. Lebret, "Applications of second-order cone programming," *Linear Algebra and its Applications*, vol. 284, no. 1, pp. 193 – 228, 1998.

[52] E. J. Candès and C. Fernandez-Granda, "Towards a mathematical theory of super-resolution," *CoRR*, vol. abs/1203.5871, 2012.

[53] Z. Yang, L. Xie, and C. Zhang, "Off-grid direction of arrival estimation using sparse bayesian inference," *IEEE Transactions on Signal Processing*, vol. 61, pp. 38–43, Jan 2013.

[54] S. Ji, Y. Xue, and L. Carin, "Bayesian compressive sensing," *IEEE Transactions on Signal Processing*, vol. 56, pp. 2346–2356, June 2008.

[55] M. E. Tipping, "Sparse Bayesian learning and the relevance vector machine," *J. Mach. Learn. Res.*, vol. 1, pp. 211–244, Sept. 2001.

[56] Z. Yang, J. Li, P. Stoica, and L. Xie, "Chapter 11 - sparse methods for direction-of-arrival estimation," in *Academic Press Library in Signal Processing, Volume 7* (R. Chellappa and S. Theodoridis, eds.), pp. 509 – 581, Academic Press, 2018.

[57] G. Tang, B. N. Bhaskar, P. Shah, and B. Recht, "Compressed sensing off the grid," *IEEE Transactions on Information Theory*, vol. 59, pp. 7465–7490, Nov 2013.

[58] Y. Li and Y. Chi, "Off-the-grid line spectrum denoising and estimation with multiple measurement vectors," *IEEE Transactions on Signal Processing*, vol. 64, pp. 1257–1269, March 2016.

[59] K. C. Toh, M. J. Todd, and R. H. Tütüncü, "SDPT3 — a Matlab software package for semidefinite programming, version 1.3," *Optimization Methods and Software*, vol. 11, no. 1-4, pp. 545–581, 1999.

[60] L. Vandenberghe and S. Boyd, "Semidefinite programming," *SIAM Review*, vol. 38, no. 1, pp. 49–95, 1996.

[61] S. Fortunati, R. Grasso, F. Gini, M. Greco, and K. Lepage, "Single-snapshot DOA estimation by using compressed sensing (JASP)," *EURASIP Journal on Advances in Signal Processing*, vol. 2014, 11 2014.

[62] C. Huang, C. Dai, T. Tsai, W. Chung, and T. Lee, "A closed-form phase-comparison ML DOA estimator for automotive radar with one single snapshot," *IEICE Electronic Express*, vol. 10, no. 7, p. 20130086, 2013.

[63] D. Malioutov, M. Cetin, and A. S. Willsky, "A sparse signal reconstruction perspective for source localization with sensor arrays," *IEEE Transactions on Signal Processing*, vol. 53, pp. 3010–3022, Aug 2005.

[64] T. B. Lavate, V. K. Kokate, and A. M. Sapkal, "Performance analysis of MUSIC and ESPRIT DOA estimation algorithms for adaptive array smart antenna in mobile communication," in *2010 Second International Conference on Computer and Network Technology*, pp. 308–311, April 2010.

[65] Y. Zhang, Z. Ye, X. Xu, and N. Hu, "Off-grid DOA estimation using array covariance matrix and block-sparse Bayesian learning," *Signal Processing*, vol. 98, pp. 197 – 201, 2014.

[66] G. Z. Karabulut, T. Kurt, and A. Yongacoglu, "Angle of arrival detection by matching pursuit algorithm," in *IEEE 60th Vehicular Technology Conference, 2004. VTC2004-Fall. 2004*, vol. 1, pp. 324–328 Vol. 1, Sep. 2004.