Spring 5-15-2015

# Discrete and Continuous Sparse Recovery Methods and Their Applications

Zhao Tan
*Washington University in St. Louis*

Washington University in St. Louis

School of Engineering and Applied Science

Department of Electrical & Systems Engineering

Dissertation Examination Committee:
Arye Nehorai, Chair
R. Martin Arthur
Viktor Gruev
Jr-Shin Li
Heinz Schäettler

# Discrete and Continuous Sparse Recovery Methods and Their Applications

by

Zhao Tan

A dissertation presented to the Graduate School of Arts and Sciences
of Washington University in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

May 2015
Saint Louis, Missouri

# Contents

# List of Figures

# List of Tables

# Acknowledgments

I would like to express my deep and sincere gratitude to my advisor, Dr. Arye Nehorai, for his professional mentoring, guidance, and support during my five years' research and life at Washington University. He has provided me a stimulating environment and also plenty of freedom to pursue the research topics that I feel interested in.

I wish to thank my dissertation defense committee members, Dr. R. Martin Arthur, Dr. Viktor Gruev, Dr. Jr-Shin Li, and Dr. Heinz Schäettler, for their valuable comments on improving my dissertation.

I further thank Dr. Yonina C. Eldar, Dr. Amir Beck, and my labmate, Peng Yang, for collaborating on this exciting research. I'm also thankful to my labmates Gongguo, Tao, Sandeep, Phani, Elad, Xiaoxiao, Keyong, Alex, Jichuan, Mengxue, Mianzhi, and Prateek who have always been supportive and helpful, and made my journey at Washington University a meaningful memory.

I owe my deepest thanks to my lovely wife Xianchao Zeng, my parents, Qingyi Tan and Qin Xu, for their patience, trust, and endless love throughout these past years.

Zhao Tan

*Washington University in Saint Louis*
*May 2015*

Dedicated to my parents and wife.

ABSTRACT OF THE DISSERTATION

Discrete and Continuous Sparse Recovery Methods and Their Applications

by

Zhao Tan

Doctor of Philosophy in Electrical Engineering

Washington University in St. Louis, 2015

Professor Arye Nehorai, Chair

Low dimensional signal processing has drawn an increasingly broad amount of attention in the past decade, because prior information about a low-dimensional space can be exploited to aid in the recovery of the signal of interest. Among all the different forms of low dimensionality, in this dissertation we focus on the synthesis and analysis models of sparse recovery. This dissertation comprises two major topics. For the first topic, we discuss the synthesis model of sparse recovery and consider the dictionary mismatches in the model. We further introduce a continuous sparse recovery to eliminate the existing off-grid mismatches for DOA estimation. In the second topic, we focus on the analysis model, with an emphasis on efficient algorithms and performance analysis.

In considering the sparse recovery method with structured dictionary mismatches for the synthesis model, we exploit the joint sparsity between the mismatch parameters and original sparse signal. We demonstrate that by exploiting this information, we can obtain a robust reconstruction under mild conditions on the sensing matrix. This model is very useful for radar and passive array applications. We propose several efficient algorithms to solve the

joint sparse recovery problem. Using numerical examples, we demonstrate that our proposed algorithms outperform several methods in the literature. We further extend the mismatch model to a continuous sparse model, using the mathematical theory of super resolution. Statistical analysis shows the robustness of the proposed algorithm. A number-detection algorithm is also proposed for the co-prime arrays. By using numerical examples, we show that continuous sparse recovery further improves the DOA estimation accuracy, over both the joint sparse method and also MUSIC with spatial smoothing.

In the second topic, we visit the corresponding analysis model of sparse recovery. Instead of assuming a sparse decomposition of the original signal, the analysis model focuses on the existence of a linear transformation which can make the original signal sparse. In this work we use a monotone version of the fast iterative shrinkage- thresholding algorithm (MFISTA) to yield efficient algorithms to solve the sparse recovery. We examine two widely used relaxation techniques, namely smoothing and decomposition, to relax the optimization. We show that although these two techniques are equivalent in their objective functions, the smoothing technique converges faster than the decomposition technique. We also compute the performance guarantee for the analysis model when a LASSO type of reconstruction is performed. By using numerical examples, we are able to show that the proposed algorithm is more efficient than other state of the art algorithms.

# Chapter 1

# Introduction

Low-dimensional signal recovery exploits the fact that many natural signals are inherently low dimensional, although they may have high ambient dimensions. Prior information about the low-dimensional space can be exploited to aid in the recovery of the signal of interest. Sparsity is one of the popular forms of prior information, and is the prior that underlies the growing field of compressive sensing [1]-[4]. This field has become a center of research interest in the areas of applied mathematics, computer science, and electrical engineering during the past decade. Compressed sensing enables signal reconstruction by using a sample rate less than the normal Nyquist rate, as long as the signal of interest is sparse in a basis representation. Compressed sensing covers a wide range of applications, such as imaging [5], radar signal processing [6]-[8], and remote sensing [9]. The research on sparse recovery and compressed sensing has mainly pursued the following two directions.

**Theoretical analysis and performance bound:** The main goal of theoretical analysis is to find the conditions under which the reconstruction is unique and exact under noiseless conditions. For noisy conditions, performance bounds are derived based on the sparsity of the original signal and the properties of the dictionary. There are two popular property measures for the dictionary: one is called the restricted isometric property, and the other is based on the coherence of the atoms in the dictionary. By performing theoretical analysis, researchers are able to find the sufficient number of samples that guarantees a robust reconstruction.

**Efficient algorithms:** Several types of efficient algorithms are explored in the literature. The first type is the greedy algorithms, such as iterative hard thresholding [10] and orthogonal matching pursuit [11], and they solve the $\ell_0$ norm by finding one nonzero term at each time. A second type of algorithms uses convex relaxation and replace the nonconvex term by a

convex approximation. In this category, we have basis pursuit [12] and LASSO [13]. Efficient algorithms, such as FISTA [14] and NESTA [15], have been proposed to solve the convex optimization. More recently, Bayesian compressive sensing theory has been proposed to further enforce the sparsity of the reconstructed signal. The idea of Bayesian compressive sensing is based on the Relevance Vector Machine (RVM) [16]. Independent Gaussian priors are assigned to each element of the sparse vector to enforce sparsity. The authors of [17] extended the RVM and showed its effectiveness in dealing with an overcomplete dictionary in the synthesis model.

In this work, we will cover fundamental theoretical research on sparse recovery and compressed sensing. In particular, we will first discuss the effect of structure dictionary mismatches in compressed sensing, and then propose discrete and continuous approaches to estimate the dictionary mismatches. In the second topic we will discuss the analysis model of sparse recovery.

## 1.1 Background

### 1.1.1 Discrete and continuous sparse recovery for compressed sensing

A typical compressed sensing problem employs the following linear model:

$$\boldsymbol{y} = \boldsymbol{D}\boldsymbol{s} + \boldsymbol{w},$$

in which $\boldsymbol{D} \in \mathbb{R}^{M\times N}(M \leq N)$ is a given dictionary matrix, $\boldsymbol{y} \in \mathbb{R}^M$ is the measurement vector, and $\boldsymbol{w} \in \mathbb{R}^M$ is the unknown noise term. We normally have $M < N$, which means the linear system is underdetermined and has infinite solutions. The signal of interest is $\boldsymbol{s} \in \mathbb{R}^N$, which is known to be sparse, i.e., the number of nonzero terms in $\boldsymbol{s}$ is far less than $N$. With this sparse prior information, we are able to reconstruct $\boldsymbol{s}$ from the underdetermined linear system when the dictionary $\boldsymbol{D}$ is known. In real applications, we normally do not have perfect information about the dictionary matrix $\boldsymbol{D}$. The dictionary can be written as $\boldsymbol{D} = \boldsymbol{A} + \boldsymbol{E}$ with matrix $\boldsymbol{A} \in \mathbb{R}^{M\times N}$ known, and matrix $\boldsymbol{E} \in \mathbb{R}^{M\times N}$ unknown. In [18],

[19], the authors showed that the reconstruction error increases with the mismatch level. In [20, 21], the alternating minimization method is proposed to solve simultaneously for the sparse signal $\boldsymbol{s}$ and mismatch $\boldsymbol{E}$. However, this method suffers from slow convergence and has no performance guarantee. One particular application for this mismatch model is DOA estimation with off-grid targets. For passive arrays and collocated MIMO radar, the sensing matrix $\boldsymbol{A}$ will be a discrete Fourier operator, and $\boldsymbol{E}$ can be approximated by $\boldsymbol{A}$'s first-order derivative. Instead of considering discrete Fourier transformation and using its first order Taylor expansion, recent studies [22] on super resolution have addressed the continuous sparse signal reconstruction with a continuous Fourier operator. By extending this approach, we will be able to draw insights into a special form of linear arrays, namely, co-prime arrays.

### 1.1.2 The analysis model of sparse recovery

Sparse recovery has two models in the current literature, the synthesis model and the analysis model. In the synthesis model of sparse recovery, it is assumed that the signal of interest can be expressed as a sparse combination of known dictionary elements. The main methods for solving the synthesis model can be classified into two categories. One includes greedy methods, such as iterative hard thresholding [10] and orthogonal matching pursuit [11]. The other is based on relaxation-type methods, such as basis pursuit [23] and LASSO [13]. However, the performance of the synthesis model deteriorates dramatically when the coherence in the dictionary increases. Recently, an alternative model has been proposed, which is known as the analysis model or co-sparse model [24], [25]. In the analysis model, instead of assuming that the signal can be decomposed as a sparse combination of a given dictionary, we assume that there exists a deterministic linear transformation which will yield a sparse signal after being applied on the original signal. Due to its importance, this counterpart has received increasing attention in recent years, and preliminary results [26]-[29] show promising performance compared with the synthesis model. Despite the recent attention, there still remains a gap between the co-sparse model and the original synthesis model with respect to efficient algorithms, performance analysis, and dictionary learning. What's more, the analysis model is generally harder than the synthesis model to solve numerically, and the

performance analysis of convex relaxation requires more effort than the previous synthesis model.

## 1.2 Contributions of this work

This dissertation first considers sparse recovery with structured mismatches in the dictionary, and also proposes a continuous sparse recovery method to deal with the off-grid effect in DOA estimation. Then we extend sparse recovery to the analysis model, develop efficient algorithm and conduct performance analysis. We summarize the main contributions as follows.

**Sparse recovery methods with structured dictionary mismatches:** In traditional compressed sensing theory, the dictionary matrix is given a priori, whereas in real applications this matrix suffers from random noise and fluctuations. In this work we consider a signal model where each column in the dictionary matrix is affected by a structured noise. This formulation is common in direction-of-arrival (DOA) estimation of off-grid targets, encountered in both radar systems and array processing. We propose to use joint sparse signal recovery to solve the compressed sensing problem with structured dictionary mismatches and also give an analytical performance bound on this joint sparse recovery. We show that, under mild conditions, the reconstruction error of the original sparse signal is bounded by both the sparsity and the noise level in the measurement model. Moreover, we implement fast first-order algorithms to speed up the computing process. Numerical examples demonstrate the good performance of the proposed algorithm, and also show that the joint-sparse recovery method yields a better reconstruction result than existing methods. By implementing the joint sparse recovery method, the accuracy and efficiency of DOA estimation are improved in both passive and active sensing cases.

**Continuous sparse recovery for DOA estimation with co-prime arrays:** In this topic, we consider the problem of direction of arrival (DOA) estimation using a recently proposed structure of non-uniform linear arrays, referred to as co-prime arrays. By exploiting the second order statistical information of the received signals, co-prime arrays exhibit $O(MN)$ degrees of freedom with only $M + N$ sensors. A sparsity-based recovery algorithm is proposed to fully utilize these degrees of freedom. The suggested method is based on the developing theory of super resolution, which considers a continuous range of possible sources

instead of discretizing this range onto a grid. With this approach, off-grid effects inherent in traditional sparse recovery can be neglected, thus improving the accuracy of DOA estimation. We show that in the noiseless case it is theoretically possible to detect up to $\frac{MN}{2}$ sources with only $2M + N$ sensors. The noise statistics of co-prime arrays are also analyzed to demonstrate the robustness of the proposed optimization scheme. A source number detection method is presented based on the spectrum reconstructed from the sparse method. By extensive numerical examples, we show the superiority of the suggested algorithm in terms of DOA estimation accuracy, degrees of freedom, and resolution ability over previous techniques, such as MUSIC with spatial smoothing and discrete sparse recovery.

**Smoothing and decomposition for analysis model of sparse recovery:** In this work, we consider algorithms and recovery guarantees for the analysis sparse model in which the signal is sparse with respect to a highly coherent frame. We consider the use of a monotone version of the fast iterative shrinkage-thresholding algorithm (MFISTA) to solve the analysis sparse recovery problem. Since the proximal operator in MFISTA does not have a closed-form solution for the analysis model, it cannot be applied directly. Instead, we examine two alternatives based on smoothing and decomposition transformations that relax the original sparse recovery problem, and then implement MFISTA on the relaxed formulation. We refer to these two methods as smoothing-based and decomposition-based MFISTA. We analyze the convergence of both algorithms, and establish that smoothing-based MFISTA converges more rapidly when applied to general nonsmooth optimization problems. We then derive a performance bound on the reconstruction error using these techniques. The bound proves that our methods can recover a signal sparse in a redundant tight frame when the measurement matrix satisfies a properly adapted restricted isometry property. Numerical examples demonstrate the performance of our methods and show that smoothing-based MFISTA converges faster than the decomposition-based alternative in real applications, such as MRI image reconstruction.

## 1.3 Organization of the dissertation

The rest of the dissertation is organized as follows. Chapter 2 considers the dictionary mismatches in the sparse recovery method, and we propose to use the joint sparsity between mismatch parameters and the original sparse signal to increase the reconstruction accuracy. In Chapter 3 we extend discrete sparse recovery to a continuous domain and propose a semidefinite programming scheme to solve the continuous optimization problem to further increase the reconstruction accuracy. Chapter 4 presents the analysis model of sparse recovery, where we consider the case that a sparse signal can be generated after linear transformation of the original signal. Efficient algorithms are proposed in this chapter, with rigorous analysis. We finally summarize the dissertation in Chapter 5, and also point out potential future directions.

## 1.4 Notations

We use a capital italic bold letter to represent a matrix and a lowercase italic bold letter to represent a vector. For a given matrix $\boldsymbol{D}$, $\boldsymbol{D}^{\mathrm{T}}, \boldsymbol{D}^{*}, \boldsymbol{D}^{\mathrm{H}}$ denote the transpose, conjugate transpose and conjugate without transpose of $\boldsymbol{D}$ respectively. We denote by $\boldsymbol{D}^{*}_{\mathcal{T}}$ the matrix that maintains the rows in $\boldsymbol{D}^{*}$ with indices in set $\mathcal{T}$, while setting all other rows to zero.

For a given vector $\boldsymbol{x}$, $\|\boldsymbol{x}\|_1, \|\boldsymbol{x}\|_2$ are the $\ell_1$ and $\ell_2$ norms, respectively, and $\|\boldsymbol{x}\|_\infty$ denotes the element in $\boldsymbol{x}$ with the largest absolute value. Let $\|\boldsymbol{x}\|_0$ represent the number of nonzero components in a vector, which is referred as the $\ell_0$ norm. Let $|\boldsymbol{x}|$ represent a vector consisting of the absolute value of every element in $\boldsymbol{x}$. $x_i$ and $x[i]$ are both used to represent the $i$th element of $\boldsymbol{x}$. We use $\odot$ to denote the point-wise multiplication of two vectors with the same dimension. We use $\otimes$ to denote the Kronecker product of two matrices. In this work, we refer a vector $\boldsymbol{s}$ as $K$-sparse if there are at most $K$ nonzero terms in $\boldsymbol{s}$. We say a vector $\boldsymbol{x} \in \mathbb{R}^{2N}$ is $K$ joint-sparse if $\boldsymbol{x} = [\boldsymbol{s}^{\mathrm{T}}, \boldsymbol{p}^{\mathrm{T}}]^{\mathrm{T}}$, with $\boldsymbol{s} \in \mathbb{R}^N$ and $\boldsymbol{p} \in \mathbb{R}^N$, both being $K$ sparse with the same support set. Then we use $\|\boldsymbol{x}\|_{0,1}$ to denote the joint sparsity of vector $\boldsymbol{x}$, and we have $\|\boldsymbol{x}\|_{0,1} = K$ at this case.

For a matrix or an operator $\boldsymbol{A}$, $\|\boldsymbol{A}\|_2$ is the induced spectral norm, and $\|\boldsymbol{A}\|_{p,q} = \max \frac{\|\boldsymbol{A}\boldsymbol{x}\|_p}{\|\boldsymbol{x}\|_q}$. For a given operator $\boldsymbol{F}$, $\boldsymbol{F}^*$ denotes the conjugate operator of $\boldsymbol{F}$. Given a function $f$, $\|f\|_{L_1}, \|f\|_{L_2}, \|f\|_{L_\infty}$ are its $\ell_1, \ell_2, \ell_\infty$ norms. Finally, $\mathrm{Re}\langle \boldsymbol{a}, \boldsymbol{b}\rangle = \frac{\langle \boldsymbol{a},\boldsymbol{b}\rangle + \langle \boldsymbol{b},\boldsymbol{a}\rangle}{2}$. We use $\mathrm{argmin}\{f(\boldsymbol{x}) : \boldsymbol{x} = \boldsymbol{z}, \boldsymbol{y}\}$ to denote $\boldsymbol{z}$ or $\boldsymbol{y}$, whichever yields a smaller function value of $f(\boldsymbol{x})$.

# Chapter 2

# Sparse Recovery Methods with Structured Dictionary Mismatches

We will discuss discrete and continuous sparse recovery methods in the following two chapters. In this chapter, we consider the problem of compressed sensing with dictionary mismatches and we apply the theoretical result to the applications of MIMO radar and nested arrays.[1]

## 2.1 Introduction

Let us recall the dictionary mismatch model for compressed sensing:

$$\boldsymbol{y} = (\boldsymbol{A} + \boldsymbol{E})\boldsymbol{s} + \boldsymbol{w}, \tag{2.1}$$

in which $\boldsymbol{A} \in \mathbb{R}^{M\times N}(M \leq N)$ is a given dictionary matrix, $\boldsymbol{y} \in \mathbb{R}^M$ is the measurement vector, and $\boldsymbol{w} \in \mathbb{R}^M$ is the unknown noise term. Matrix $\boldsymbol{E} \in \mathbb{R}^{M\times N}$ is the unknown mismatch of the dictionary. The signal of interest is $\boldsymbol{s} \in \mathbb{R}^N$, which is known to be sparse, i.e., the number of nonzero terms in $\boldsymbol{s}$ is far less than $N$. In this chapter, we consider a particular structured dictionary mismatch model with $\boldsymbol{d}_i = \boldsymbol{a}_i + \beta_i \boldsymbol{b}_i, 1 \leq i \leq N$, where $\boldsymbol{d}_i$ and $\boldsymbol{a}_i$ are the $i$-th column of matrices $\boldsymbol{D}$ and $\boldsymbol{A}$ respectively; $\boldsymbol{a}_i$ and $\boldsymbol{b}_i$ are given for all $i$,

[1]This chapter is based on Z. Tan, P. Yang, and A. Nehorai, "Joint sparse recovery method for compressed sensing with structured dictionary mismatch," *IEEE Trans. Signal Processing*, Vol. 62, pp. 4997-5008, Oct. 2014. 

and $\beta_i$ is unknown. Thus the signal model in this chapter is

$$\boldsymbol{y} = (\boldsymbol{A} + \boldsymbol{B}\boldsymbol{\Delta})\boldsymbol{s} + \boldsymbol{w}, \tag{2.2}$$

where $\boldsymbol{\Delta} = \text{diag}(\boldsymbol{\beta})$, $\boldsymbol{\beta} = [\beta_1, \beta_2, \ldots, \beta_N]^{\text{T}}$, and $\boldsymbol{B} = [\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_N] \in \mathbb{R}^{M \times N}$.

This structured mismatch was previously considered in [20, 30]. Although it is a limited mismatch model, it has many applications in areas such as spectral estimation, radar signal processing, and DOA estimation. In [31], a greedy method based on matching pursuit is proposed to combine with the total least square method to deal with the structured mismatch for compressed sensing. In [20, 30], a bounded mismatch parameter $\boldsymbol{\beta}$ is considered, which is common in DOA estimations for off-grid targets. The proposed frameworks were based on the first order Taylor expansion, and they enforced the sparsity of the original signal $\boldsymbol{s}$. They were solved using interior point methods [32], which require solving linear systems, and the computing speed can be extremely slow when the problem's dimension grows.

In this work, we first propose to use the idea of the joint-sparse recovery [33],[34] to further exploit the underlying structure in compressed sensing with the structured dictionary mismatch. Joint sparsity in this chapter indicates that the nonzero terms in the sparse signal come in pairs. We also give a performance guarantee when the sensing matrix $\boldsymbol{A}$ and the mismatch matrix $\boldsymbol{B}$ satisfy certain constraints. For large-dimensional problems, we implement the idea of a first-order algorithm, named fast iterative shrinkage-thresholding algorithm (FISTA) [14], to solve the joint-sparse recovery with both bounded and unbounded mismatch parameter $\beta$. FISTA is a special case of a general algorithmic framework [35] and is more efficient in dealing with large dimensional data than the interior point methods. Some preliminary results of this work were shown in [36].

We extend the developed theory and algorithms to real DOA estimation applications with both passive and active sensing. Since the number of targets in the region of interest is limited, DOA estimation benefits from compressed sensing: both sampling energy and processing time can be greatly reduced. In order to implement compressed sensing, the region of interest needs to be discretized into a grid. The existence of off-grid targets deteriorates the performance of compressed sensing dramatically. Recent research has used compressed sensing in both active sensing application [6]-[8] and passive sensing [37, 38]. However, none of these works consider the situation of off-grid targets. According to the numerical example

shown in this chapter, by exploiting the first order derivative of sensing model associated with off-grid targets and also the joint sparsity between original signal and mismatch parameter, the accuracy of DOA estimation can be improved compared with previous methods.

The chapter is organized as follows. In section 2.2 we introduce the model for compressed sensing with structured dictionary mismatches and propose to use joint sparsity to solve the reconstruction problem. We analyze the performance bound on the reconstruction error using the proposed joint sparse recovery method. In section 2.3 we extend the general mismatch models to the research area of DOA estimation with off-grid targets and we also describe the mathematical model for both passive sensing and active sensing applications with off-grid targets.. In section 2.4, we give the FISTA implementation of the joint sparse recovery methods. In section 2.5, we use several numerical examples to demonstrate that the proposed method outperforms existing methods for compressed sensing with structured dictionary mismatches.

## 2.2 General structured dictionary model

### 2.2.1 Compressed sensing with dictionary mismatches

Traditional compressed sensing can be solved using the LASSO formulation [13], stated as

$$\text{(LASSO)} \quad \min_{\boldsymbol{s}\in\mathbb{R}^n} \frac{1}{2}\|\boldsymbol{D}\boldsymbol{s}-\boldsymbol{y}\|_2^2+\lambda\|\boldsymbol{s}\|_1. \tag{2.3}$$

In order to recover the sparse signal $\boldsymbol{s}$ in the mismatch model (2.2), having $\boldsymbol{D}=\boldsymbol{A}+\boldsymbol{B}\boldsymbol{\Delta}$ the optimization problem is given as

$$\min_{\boldsymbol{s}\in\mathbb{R}^N,\boldsymbol{\beta}\in\mathbb{R}^N} \frac{1}{2}\|(\boldsymbol{A}+\boldsymbol{B}\boldsymbol{\Delta})\boldsymbol{s}-\boldsymbol{y}\|_2^2+\lambda\|\boldsymbol{s}\|_1, \text{s.t.}\boldsymbol{\Delta}=\text{diag}(\boldsymbol{\beta}). \tag{2.4}$$

The above optimization is non-convex and generally hard to solve. Please note that when $s_i=0$ for certain $i$, then $\beta_i$ can be any value, without affecting the reconstruction. Therefore, in the rest of this chapter, we focus only on instances of $\beta_i$ with nonzero $s_i$. In [20, 21], the authors proposed to use the alternating minimization method to solve for both $\boldsymbol{s}$ and $\boldsymbol{\beta}$ when

the mismatch variable $\boldsymbol{\beta}$ is bounded or Gaussian distributed. Based on the idea of [20], we let $\boldsymbol{p} = \boldsymbol{\beta} \odot \boldsymbol{s}$ and $\boldsymbol{\Phi} = [\boldsymbol{A}, \boldsymbol{B}]$, and then transform the original non-convex optimization into a relaxed convex one. Due to the fact that $p_i$ is zero whenever $s_i$ is zero, instead of enforcing the sparsity of $\boldsymbol{s}$ in [20, 30] we enforce the joint sparsity between $\boldsymbol{s}$ and $\boldsymbol{p}$. We let $\boldsymbol{x} = [\boldsymbol{s}^{\mathrm{T}}, \boldsymbol{p}^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{2N}$, and define the mixed $\ell_2/\ell_1$ norm of $\boldsymbol{x}$ as

$$\|\boldsymbol{x}\|_{2,1} = \sum_{i=1}^{N} \sqrt{x_i^2 + x_{N+i}^2}. \tag{2.5}$$

Also we define

$$\|\boldsymbol{x}\|_{\infty,1} = \max_{1 \le i \le N} \sqrt{x_i^2 + x_{N+i}^2}. \tag{2.6}$$

If $\boldsymbol{s}$ is $K$-sparse, then $\boldsymbol{p}$ will also be $K$-sparse, with the same support set as $\boldsymbol{s}$. Hence the relaxed optimization enforcing joint sparsity will be referred as (JS) throughout the chapter and it can be stated as

$$\text{(JS)} \quad \min_{\boldsymbol{x} \in \mathbb{R}^{2N}} \frac{1}{2} \|\boldsymbol{\Phi}\boldsymbol{x} - \boldsymbol{y}\|_2^2 + \lambda \|\boldsymbol{x}\|_{2,1}. \tag{2.7}$$

### 2.2.2 Performance bound for joint sparse LASSO

In order to analyze the recovery performance of (JS), we introduce the joint restricted isometry property (J-RIP), similar to the restricted isometry property (RIP) [1] in compressed sensing. This definition is a special case of the Block RIP introduced in [33].

**Definition 2.1.** ***(J-RIP)*** *We say that the measurement matrix $\boldsymbol{\Phi} \in \mathbb{R}^{M \times 2N}$ obeys the joint restricted isometry property with constant $\sigma_K$ if*

$$(1 - \sigma_K)\|\boldsymbol{v}\|_2^2 \le \|\boldsymbol{\Phi}\boldsymbol{v}\|_2^2 \le (1 + \sigma_K)\|\boldsymbol{v}\|_2^2 \tag{2.8}$$

*holds for all $K$ joint-sparse vectors $\boldsymbol{v} \in \mathbb{R}^{2N}$.*

With this definition a non-convex recovery scheme can be obtained.

**Theorem 2.1.** *Let $\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{x}$, and $\boldsymbol{\Phi} \in \mathbb{R}^{M \times 2N}$, $\boldsymbol{x} = [\boldsymbol{s}^{\mathrm{T}}, \boldsymbol{p}^{\mathrm{T}}]^{\mathrm{T}}$, in which $\boldsymbol{p} = \boldsymbol{s} \odot \boldsymbol{\beta} \in \mathbb{R}^N$ and $\boldsymbol{s} \in \mathbb{R}^N$. Let $\|\boldsymbol{x}\|_{0,1}$ denote the joint sparsity of vector $\boldsymbol{x}$. Assume the matrix $\boldsymbol{\Phi}$ satisfies the*

*J-RIP condition with constant $\sigma_{2K} < 1$ and $\boldsymbol{s}$ has at most $K$ nonzero terms. By solving the following non-convex optimization problem*

$$\min_{\boldsymbol{x}\in\mathbb{R}^{2N}} \|\boldsymbol{x}\|_{0,1}, \quad \text{s.t.} \quad \boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{x}, \tag{2.9}$$

*we obtain the optimal solution $\hat{\boldsymbol{x}}$. Then $s_i = \hat{x}_i$ for all $i$, and $\beta_i = \hat{x}_{N+i}/\hat{x}_i$ when $s_i$ is nonzero.*

**Proof:** When $\boldsymbol{s}$ has sparsity $K$, then we know that $\|\boldsymbol{x}\|_{0,1} \leq K$. Then since $\hat{\boldsymbol{x}}$ solves the optimization problem, we have $\|\hat{\boldsymbol{x}}\|_{0,1} \leq \|\boldsymbol{x}\|_{0,1} \leq K$, and then $\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_{0,1} \leq 2K$. Since both $\hat{\boldsymbol{x}}$ and $\boldsymbol{x}$ meet the equality constraint, we have $\boldsymbol{\Phi}\boldsymbol{x} = \boldsymbol{y}$ and $\boldsymbol{\Phi}\hat{\boldsymbol{x}} = \boldsymbol{y}$, thus $\boldsymbol{\Phi}(\boldsymbol{x} - \hat{\boldsymbol{x}}) = 0$. Using the property of J-RIP, we have

$$(1 - \sigma_{2K})\|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_2^2 \leq \|\boldsymbol{\Phi}(\boldsymbol{x} - \hat{\boldsymbol{x}})\|_2^2 = 0. \tag{2.10}$$

Hence we have $\hat{\boldsymbol{x}} = \boldsymbol{x} = [\boldsymbol{s}^{\mathrm{T}}, \boldsymbol{p}^{\mathrm{T}}]^{\mathrm{T}}$. Since $\boldsymbol{p} = \boldsymbol{s} \odot \boldsymbol{\beta}$, we than obtain $\boldsymbol{s}$ and $\boldsymbol{\beta}$ from $\hat{\boldsymbol{x}}$. □

Since the above optimization is non-convex, the $\ell_{2,1}$ norm is used instead of the joint sparsity. Considering the noise in the signal model, the optimization takes the form

$$\min_{\boldsymbol{x}\in\mathbb{R}^{2N}} \|\boldsymbol{x}\|_{2,1}, \quad \text{s.t.} \quad \|\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{x}\| \leq \varepsilon. \tag{2.11}$$

The (JS) is equivalent to the above formulation, i.e., for a given $\varepsilon$, there is a $\lambda$ that makes these two optimizations yield the same optimal point. A theoretical guarantee for (2.11) is given in [33], however this result cannot be directly applied to (JS). A performance bound for (JS) can be obtained based on techniques introduced in [33, 39] and [40], and is given in the following theorem. The details of the proof is included in the Appendix A.

**Theorem 2.2.** *Let $\boldsymbol{\Phi} \in \mathbb{R}^{M\times 2N}$ satisfy the joint RIP with $\sigma_{2K} < 0.1907$. Let the measurement $\boldsymbol{y}$ follow $\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{x} + \boldsymbol{w}$, where $\boldsymbol{w}$ is the measurement noise in the linear system. Assume that $\lambda$ obeys $\|\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{w}\|_{\infty,1} \leq \frac{\lambda}{2}$, and then the solution $\hat{\boldsymbol{x}}$ to the optimization problem (JS) satisfies*

$$\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2 \leq C_0\sqrt{K}\lambda + C_1\frac{\|\boldsymbol{x} - (\boldsymbol{x})_K\|_{2,1}}{\sqrt{K}}. \tag{2.12}$$

*Here $(\boldsymbol{x})_K$ is the best $K$ joint-sparse approximation to $\boldsymbol{x}$. $C_0$ and $C_1$ are constants that depend on $\sigma_{2K}$.*

**Remarks:**

**1.** In [33], it was shown that random matrices satisfy the J-RIP with an overwhelming probability, and this probability is much larger than the probability of satisfying the traditional RIP under the same circumstance.

**2.** In our case, $\boldsymbol{x} = [\boldsymbol{s}^{\mathrm{T}}, \boldsymbol{p}^{\mathrm{T}}]^{\mathrm{T}}$. So if $\boldsymbol{s}$ is $K$-sparse, since $p = \boldsymbol{\beta} \odot \boldsymbol{s}$, then $\boldsymbol{x}$ will be joint $K$-sparse. Thus we have $\|\boldsymbol{x} - (\boldsymbol{x})_K\|_{2,1} = 0$, and the reconstruction error depends only on the noise level, which is characterized by $\lambda$.

**3.** In the performance bound (2.12), the bound is on the reconstruction error of $\boldsymbol{x}$, while we care more about the error bound of $\boldsymbol{s}$. It is easy to get

$$\|\hat{\boldsymbol{s}} - \boldsymbol{s}\|_2 \leq \|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2 \leq C_0\sqrt{K}\lambda + C_1\frac{\|\boldsymbol{x} - (\boldsymbol{x})_K\|_{2,1}}{\sqrt{K}}. \tag{2.13}$$

**4.** In some applications, we care about $\beta_i$ only when the signal $s_i$ is nonzero. For the $i$-th element of the mismatch variable $\boldsymbol{\beta}$, we have

$$|\hat{\beta}_i\hat{s}_i - \beta_i s_i| \leq C, \tag{2.14}$$

where $C = C_0\sqrt{K}\lambda + C_1\frac{\|\boldsymbol{x}-(\boldsymbol{x})_K\|_{2,1}}{\sqrt{K}}$. Using triangle inequality, we have

$$|\hat{s}_i||\beta_i - \hat{\beta}_i| \leq C + |\beta_i||s_i - \hat{s}_i|. \tag{2.15}$$

When $s_i$ is nonzero, the reconstructed $\hat{s}_i$ is also highly likely to be nonzero, which is confirmed by numerical examples. In real applications, the mismatch term $\beta$ is often bounded; therefore, we can bound the reconstruction error of $\beta_i$ as

$$|\beta_i - \hat{\beta}_i| \leq \frac{C + |\beta_i||s_i - \hat{s}_i|}{|\hat{s}_i|}. \tag{2.16}$$

**5.** There are two ways to recover the mismatch parameter $\boldsymbol{\beta}$. The first way is to directly use the optimal solution from solving (JS) and let $\hat{\beta}_i = \hat{p}_i/\hat{s}_i$. The other way is to use the recovered $\hat{\boldsymbol{s}}$ from solving (JS) and plug it back in the original optimization problem (2.4) to solve for $\boldsymbol{\beta}$.

## 2.3 DOA estimation with off-grid targets

### 2.3.1 Off-grid compressed sensing

We begin by introducing the general model encountered in DOA estimation, which is also referred as the translation-invariant model in [30]. The $m$th measurement in the model is described by

$$y_m = \sum_{k=1}^{K} f_k a_m(\tau_k) + w_m, \tag{2.17}$$

where $\tau_k$ is the location of $k$th target, $w_m$ is the measurement noise and $f_k$ is the signal transmitted from $k$th target. Suppose that the region of interest spans from $\theta_1$ to $\theta_N$. Then the traditional approach is via discretizing the continuous region uniformly into a grid such as $\boldsymbol{\theta} = [\theta_1, \theta_2, \ldots, \theta_N]$ with step size $2r$, i.e., $\theta_{i+1} - \theta_i = 2r, 1 \leq i \leq N-1$. Thus the signal model can be written as

$$\boldsymbol{y} = \boldsymbol{A}(\boldsymbol{\theta})\boldsymbol{s} + \boldsymbol{w}, \tag{2.18}$$

where $A_{mn}(\boldsymbol{\theta}) = a_m(\theta_n)$, and $\boldsymbol{w} = [w_1, w_2, \ldots, w_M]^{\mathrm{T}}$ is the noise term. $s_n$ is equal to $f_k$ when $\theta_n = \tau_k$ for certain $k$, otherwise $s_n$ is zero.

The model (2.18) is accurate only when $\tau_k \in \boldsymbol{\theta}$ for all $k$. When the actual parameters do not fall exactly on the discretized grid $\boldsymbol{\theta}$, the modeling error deteriorates the reconstruction accuracy, and the performance of compressed sensing can be highly jeopardized [18]. Let $\boldsymbol{\varphi} = [\varphi_1, \varphi_2, \ldots, \varphi_N]$ be the unknown grid, such that $\tau_k \in \boldsymbol{\varphi}$ for all $k$, and $|\varphi_n - \theta_n| \leq r$ with $1 \leq n \leq N$. In this chapter, we assume that two targets are at least $2r$ apart, i.e., $|\tau_i - \tau_j| > 2r$ for all $1 \leq i, j \leq K$. Using the first order Taylor expansion, a more accurate signal model can be described by the unknown grid $\boldsymbol{\varphi}$ as

$$\boldsymbol{y} = \boldsymbol{A}(\boldsymbol{\varphi})\boldsymbol{s} + \boldsymbol{w} \approx (\boldsymbol{A} + \boldsymbol{B}\boldsymbol{\Delta})\boldsymbol{s} + \boldsymbol{w}, \tag{2.19}$$

where $\boldsymbol{A} = \boldsymbol{A}(\boldsymbol{\theta}), \boldsymbol{B} = [\frac{\partial \boldsymbol{a}(\theta_1)}{\partial \theta_1}, \frac{\partial \boldsymbol{a}(\theta_2)}{\partial \theta_2}, \ldots, \frac{\partial \boldsymbol{a}(\theta_N)}{\partial \theta_N}], \boldsymbol{\Delta} = \mathrm{diag}(\boldsymbol{\beta})$, and $\boldsymbol{\beta} = \boldsymbol{\varphi} - \boldsymbol{\theta}$. The reconstruction of the original signal $\boldsymbol{s}$ and grid mismatch $\boldsymbol{\beta}$ can be estimated by solving the (JS) optimization in (2.7).

Since we know that every element in $\boldsymbol{\beta}$ is in the range of $[-r, r]$, one more bounded constraint can be added. By letting $\boldsymbol{p} = \boldsymbol{\beta} \odot \boldsymbol{s}$ and penalizing the joint sparsity between $\boldsymbol{s}$ and $\boldsymbol{p}$ we can state the non-convex bounded joint sparse method as

$$\begin{aligned} \min_{\boldsymbol{s},\boldsymbol{p},\boldsymbol{x}} \quad & \tfrac{1}{2}\|\boldsymbol{A}\boldsymbol{s} + \boldsymbol{B}\boldsymbol{p} - \boldsymbol{y}\|_2^2 + \lambda\|\boldsymbol{x}\|_{2,1}, \\ \text{s.t.} \quad & -r|\boldsymbol{s}| \leq \boldsymbol{p} \leq r|\boldsymbol{s}|, \\ & \boldsymbol{x} = [\boldsymbol{s}^{\mathrm{T}}, \boldsymbol{p}^{\mathrm{T}}]^{\mathrm{T}}. \end{aligned} \tag{2.20}$$

The above optimization is hard to solve. However when $\boldsymbol{s}$ is a positive vector, the above optimization is convex and given as

$$\begin{aligned} \text{(BJS)} \quad \min_{\boldsymbol{s},\boldsymbol{p},\boldsymbol{x}} \quad & \tfrac{1}{2}\|\boldsymbol{A}\boldsymbol{s} + \boldsymbol{B}\boldsymbol{p} - \boldsymbol{y}\|_2^2 + \lambda\|\boldsymbol{x}\|_{2,1}, \\ \text{s.t.} \quad & -r\boldsymbol{s} \leq \boldsymbol{p} \leq r\boldsymbol{s}, \quad \boldsymbol{s} \geq 0, \\ & \boldsymbol{x} = [\boldsymbol{s}^{\mathrm{T}}, \boldsymbol{p}^{\mathrm{T}}]^{\mathrm{T}}. \end{aligned} \tag{2.21}$$

This formulation can be solved by standard convex optimization methods, such as interior point methods. When the dimension of the problem increases, a fast algorithm is implemented to reduce the computational burden, as we will illustrate later in this chapter.

### 2.3.2 Merging process for representation ambiguity

When a target is located at the midpoint of the interval $[\theta_i, \theta_{i+1}]$ with length $2r$, then the DOA of that target can be regarded as either $\theta_i + r$ or $\theta_{i+1} - r$. This phenomenon leads to ambiguity in the reconstruction. Even in cases when the target is near the midpoint of the interval $[\theta_i, \theta_{i+1}]$, due to the measurement noise we normally have two nonzero terms of the reconstructed signal located in the interval $[\theta_i, \theta_{i+1}]$.

To resolve this problem, we perform a linear interpolation on the two nonzero terms in the same interval and merge them into one target, since we know a priori that the two targets are at least $2r$ apart. Suppose that after solving (BJS) we have two recovered DOAs, $\varphi_a, \varphi_b \in [\theta_i, \theta_{i+1}]$. The corresponding reconstructed signal magnitudes are $s_a$ and $s_b$. After

merging them, we have only one recovered DOA $\varphi$, with magnitude $s$ given as

$$s = s_a + s_b, \text{ and } \varphi = \theta_c + \frac{|s_a|(\varphi_a - \theta_c) + |s_b|(\varphi_b - \theta_c)}{|s_a| + |s_b|}, \tag{2.22}$$

where $\theta_c$ is the midpoint of interval $[\theta_i, \theta_{i+1}]$.

### 2.3.3 Passive sensing: nonuniform linear arrays

The nonuniform linear array considered in this chapter consists of $L$ sensors which are linearly located. We suppose the $l$th sensor is located at $d_l$. By discretizing the range of interest as $[\theta_1, \theta_2, \ldots \theta_N]$, the received signal at time $t$ is given as

$$\boldsymbol{x}(t) = \sum_{p=1}^{P} \alpha_p(t)\boldsymbol{\phi}(\theta_p) + \boldsymbol{e}, \tag{2.23}$$

where $\alpha_p(t)$ is the signal transmitted with power $\sigma_p^2$ from the target at grid point $p$, with $\sigma_p$ equal to zero when there is no target at grid point $p$. $\boldsymbol{\phi}(\theta_p)$ is the steering vector for grid point $\theta_p$, with the $l$th element equal to $e^{\boldsymbol{\jmath}(2\pi/\lambda)d_l \sin(\theta_p)}$, and $\lambda$ is the wavelength.

We assume that all the targets are uncorrelated and that the noise is white Gaussian with noise power $\sigma_n^2$. Recent research [41, 42] has proposed analyzing the covariance matrix of $\boldsymbol{x}(t)$ to increase the degrees of freedom of the original system. The covariance matrix of $\boldsymbol{x}$ is given as

$$\boldsymbol{R}_{\boldsymbol{xx}} = E(\boldsymbol{x}\boldsymbol{x}^*) = \sum_{p=1}^{P} \sigma_p^2 \boldsymbol{\phi}(\theta_p)\boldsymbol{\phi}(\theta_p)^* + \sigma_n^2 \boldsymbol{I}, \tag{2.24}$$

in which $\boldsymbol{I}$ is an identity matrix. By vectoring the above equation, we have

$$\boldsymbol{y} = \boldsymbol{A}(\boldsymbol{\theta})\boldsymbol{s} + \sigma^2 \mathbf{1}_n, \tag{2.25}$$

where $\boldsymbol{A}(\boldsymbol{\theta}) = [\boldsymbol{\phi}(\theta_1)^{\mathrm{H}} \otimes \boldsymbol{\phi}(\theta_1), \ldots, \boldsymbol{\phi}(\theta_P)^{\mathrm{H}} \otimes \boldsymbol{\phi}(\theta_P)]$, and $\boldsymbol{s}$ is a sparse signal equaling $[\sigma_1^2, \ldots, \sigma_P^2]^{\mathrm{T}}$. We have $\mathbf{1}_n = [\boldsymbol{e}_1^{\mathrm{T}}, \boldsymbol{e}_2^{\mathrm{T}}, \ldots, \boldsymbol{e}_L^{\mathrm{T}}]^{\mathrm{T}}$, where $\boldsymbol{e}_i$ contains all zero elements except for $i$-th element, which equals one. Since $\boldsymbol{s}$ is a positive vector, the (BJS) formulation in (2.21) can be implemented with $\boldsymbol{B} = [\frac{\partial(\boldsymbol{\phi}(\theta_1)^* \otimes \boldsymbol{\phi}(\theta_1))}{\partial \theta_1}, \ldots, \frac{\partial(\boldsymbol{\phi}(\theta_P)^* \otimes \boldsymbol{\phi}(\theta_P))}{\partial \theta_P}]$.

### 2.3.4 Active sensing: MIMO radar

The MIMO radar model is based on the model introduced in [7]. To make the chapter self-contained we review the radar model in [7] and then expand it to a general model considering off-grid targets.

We consider a MIMO radar system with $M_{\mathrm{T}}$ transmitters, $M_{\mathrm{R}}$ receivers. Suppose there are $K$ targets in the area of interest. In our case, we suppose the targets are stationary or moving very slowly compared with the sampling rate of the radar system. So the Doppler effect is neglected. The locations of transmitters and receivers are randomly generated within a disk. We consider the problem in two dimensional space using polar coordinates. The location of the $i$-th transmitter is given by $[d_i^{\mathrm{t}}, \phi_i^{\mathrm{t}}]$, and the location of the $j$-th receiver by $[d_j^{\mathrm{r}}, \phi_j^{\mathrm{r}}]$. The region of interest is discretized into a grid. Suppose that the location of the $p$-th grid point is indicated by $[l_p, \theta_p]$. We assume that $l_{\mathrm{p}} \gg d_i^{\mathrm{t}}$ and $l_{\mathrm{p}} \gg d_j^{\mathrm{r}}$ for all $i, j$ and $p$. With this far field assumption, the distance between the $i$-th transmitter and the $p$-th grid point can be approximated as

$$d_{ip}^{\mathrm{t}} = l_p - \gamma_{ip}^{\mathrm{t}}, \tag{2.26}$$

where $\gamma_{ip}^{\mathrm{t}} = d_i^{\mathrm{t}} \cos(\phi_i^{\mathrm{t}} - \theta_p)$. We can also approximate the distance between the $j$-th transmitter and the $p$-th grid point as

$$d_{jp}^{\mathrm{r}} = l_p - \gamma_{jp}^{\mathrm{r}}, \tag{2.27}$$

where $\gamma_{jp}^{\mathrm{r}} = d_j^{\mathrm{r}} \cos(\phi_j^{\mathrm{r}} - \theta_p)$.

Assume the transmitted signal from $i$-th transmitter is narrow band and is given as $x_i(t) e^{\boldsymbol{j} 2\pi f_c t}$, $i = 1, ..., M_{\mathrm{T}}$. Here $f_c$ indicates the transmitting frequency of the radar signal. Then the signal received by the $p$-th grid point in the scene can be written as

$$y_p(t) = \sum_{i=1}^{M_{\mathrm{T}}} x_i(t - \tau_{ip}^{\mathrm{t}}) e^{\boldsymbol{j} 2\pi f_c (t - \tau_{ip}^{\mathrm{t}})}, \quad p = 1, ..., P, \tag{2.28}$$

where $\tau_{ip}^{\mathrm{t}}$ represents the delay between the $i$-th transmitter and the $p$-th grid point. Therefore we can write the signal received by $j$-th receiver as

$$z_j(t) = \sum_{p=1}^{P} \sum_{i=1}^{M_{\mathrm{T}}} \alpha_p x_i(t - \tau_{ip}^{\mathrm{t}} - \tau_{jp}^{\mathrm{r}}) e^{\boldsymbol{j} 2\pi f_c (t - \tau_{ip}^{\mathrm{t}} - \tau_{jp}^{\mathrm{r}})}, \quad j = 1, \ldots, M_{\mathrm{R}},$$

where $\tau_{jp}^{\mathrm{r}}$ represents the delay between the $j$-th receiver and the $p$-th grid point and $\alpha_p$ represents the refection factor if there is a target located at grid point $p$ otherwise it is zero. The term $e^{\boldsymbol{j} 2\pi f_c t}$ can also be known if the transmitters are synchronized and also share the same clock with each receivers. With the narrow band and far-field assumptions, we have

$$z_j(nT) = \sum_{p=1}^{P} \sum_{i=1}^{M_{\mathrm{T}}} \alpha_p x_i(nT) e^{-\boldsymbol{j} 2\pi f_c (\tau_{ip}^{\mathrm{t}} + \tau_{jp}^{\mathrm{r}})}, \quad j = 1, \ldots, M_R,$$

in which $T$ is the sampling interval. The delay term in the previous equations can be calculated as $\tau_{ip}^{\mathrm{t}} = d_{ip}^{\mathrm{t}}/c, \tau_{jp}^{\mathrm{r}} = d_{jp}^{\mathrm{r}}/c$,where $c$ stands for the transmission velocity of the signal.

Now we rewrite the signal model in a sampled format which is more conventionally used for a signal processing system and write it as a matrix equation. In the following equations we neglect the sample interval $T$ for simplicity. The received signal at the $p$-th grid point equals

$$y_p(n) = \sum_{i=1}^{M_{\mathrm{T}}} x_i(n) e^{-\boldsymbol{j} \frac{2\pi f_c}{c} d_{ip}^{\mathrm{t}}} = e^{-\boldsymbol{j} \frac{2\pi f_c}{c} l_p} \sum_{i=1}^{M_{\mathrm{T}}} x_i(n) e^{\boldsymbol{j} \frac{2\pi f_c}{c} \gamma_{ip}^{\mathrm{t}}}, \tag{2.29}$$

where $n$ is the time index for the $n$-th sample. After expressing equation (2.29) in its vector form, we have

$$y_p(n) = e^{-\boldsymbol{j} \frac{2\pi f_c}{c} l_p} \boldsymbol{x}^{\mathrm{T}}(n) \boldsymbol{u}_p, \tag{2.30}$$

where

$$\boldsymbol{x}(n) = [x_1(n), \cdots, x_{M_{\mathrm{T}}}(n)]^{\mathrm{T}}, \tag{2.31}$$

$$\boldsymbol{u}_p = [e^{\boldsymbol{j} \frac{2\pi f_c}{c} \gamma_{1p}^{\mathrm{t}}}, \cdots, e^{\boldsymbol{j} \frac{2\pi f_c}{c} \gamma_{M_{\mathrm{T}} p}^{\mathrm{t}}}]^{\mathrm{T}}. \tag{2.32}$$

The signal received by the j-th receiver can be expressed as

$$z_j(n) = \sum_{p=1}^{P} \alpha_p e^{-\boldsymbol{\jmath}\frac{2\pi f_c}{c} l_p} e^{\boldsymbol{\jmath}\frac{2\pi f_c}{c}\gamma_{jp}^{\mathrm{r}}} y_p(n), \quad j = 1, \ldots, M_{\mathrm{R}}. \tag{2.33}$$

Suppose we take $L$ snapshots, and then stack all the measurements from the $j$-th receiver in one vector. We will have

$$\boldsymbol{z}_j = \begin{pmatrix} z_j(0) \\ \vdots \\ z_j(L-1) \end{pmatrix} = \sum_{p=1}^{P} \alpha_p e^{-\boldsymbol{\jmath}\frac{4\pi f_c}{c} l_p} e^{\boldsymbol{\jmath}\frac{2\pi f_c}{c}\gamma_{jp}^{\mathrm{r}}} \boldsymbol{X}\boldsymbol{u}_p, \tag{2.34}$$

where $\boldsymbol{X} = [\boldsymbol{x}(0), \ldots, \boldsymbol{x}(L-1)]^{\mathrm{T}}$.

In this linear model the sparse signal $\boldsymbol{s}$ is given as

$$s_p = \begin{cases} \alpha_p e^{-\boldsymbol{\jmath}\frac{4\pi f_c}{c} l_p} & \text{if there is a target at } \theta_p, \\ 0 & \text{if there is no target.} \end{cases} \tag{2.35}$$

Considering the measuring noise in the process, the received signal collected at $j$-th receiver is described as

$$\boldsymbol{z}_j = \sum_{p=1}^{P} e^{\boldsymbol{\jmath}\frac{2\pi f_c}{c}\gamma_{jp}^{\mathrm{r}}} \boldsymbol{X}\boldsymbol{u}_p s_p + \boldsymbol{e}_j, \tag{2.36}$$

in which $\boldsymbol{e}_j$ denotes the noise received by the $j$-th receiver during sampling. In our work we assume the noise is i.i.d. Gaussian.

Then we can rewrite equation (2.36) as

$$\boldsymbol{z}_j = \sum_{p=1}^{P} e^{\boldsymbol{\jmath}\frac{2\pi f_c}{c}\gamma_{jp}^{\mathrm{r}}} \boldsymbol{X}\boldsymbol{u}_p s_p + \boldsymbol{e}_j = \boldsymbol{\Psi}_j \boldsymbol{s} + \boldsymbol{e}_j, \tag{2.37}$$

in which $\boldsymbol{s} = [s_1, \ldots, s_P]^{\mathrm{T}}$, which indicates the locational signal, and $\boldsymbol{\Psi}_j$ represents the measuring matrix for the $j$-th receiver:

$$\boldsymbol{\Psi}_j = [e^{\boldsymbol{\jmath}\frac{2\pi f_c}{c}\gamma_{j1}^{\mathrm{r}}} \boldsymbol{X}\boldsymbol{u}_1, \ldots, e^{j\frac{2\pi f_c}{c}\gamma_{jP}^{\mathrm{r}}} \boldsymbol{X}\boldsymbol{u}_P]. \tag{2.38}$$

After making all these measurements, a sensing matrix is used to reduce the dimension of the problem. For the $j$-th receiver, we have a matrix $\boldsymbol{\Phi}_j \in \mathbb{R}^{M\times L}$ which is randomly generated and also satisfies the condition that $\boldsymbol{\Phi}_j\boldsymbol{\Phi}_j^{\mathrm{T}} = \boldsymbol{I}$ and $M \leq L$ The compressed data of the $j$-th receiver is given as

$$\boldsymbol{y}_j = \boldsymbol{\Phi}_j\boldsymbol{\Psi}_j\boldsymbol{s} + \boldsymbol{\Phi}_j\boldsymbol{e}_j. \tag{2.39}$$

To make the model more concise, we stack compressed data generated by all the receivers into one vector:

$$\boldsymbol{y} = \begin{pmatrix} \boldsymbol{y}_1 \\ \vdots \\ \boldsymbol{y}_{M_{\mathrm{R}}} \end{pmatrix} = \boldsymbol{A}(\boldsymbol{\theta})\boldsymbol{s} + \boldsymbol{w}, \tag{2.40}$$

where

$$\boldsymbol{A}(\boldsymbol{\theta}) = \begin{pmatrix} \boldsymbol{\Phi}_1\boldsymbol{\Psi}_1 \\ \vdots \\ \boldsymbol{\Phi}_{M_{\mathrm{R}}}\boldsymbol{\Psi}_{M_{\mathrm{R}}} \end{pmatrix}, \boldsymbol{w} = \begin{pmatrix} \boldsymbol{\Phi}_1\boldsymbol{e}_1 \\ \vdots \\ \boldsymbol{\Phi}_{M_{\mathrm{R}}}\boldsymbol{e}_{M_{\mathrm{R}}} \end{pmatrix}. \tag{2.41}$$

However, in real applications the targets' locations does not fall exactly on the grid point chosen to perform compressed sensing. According to the idea introduced in section 2.3.1, suppose the actual non-uniform grid we want to use is $\boldsymbol{\varphi} = [\varphi_1, \ldots, \varphi_P]^{\mathrm{T}}$, and we need to take $\boldsymbol{\beta} = \boldsymbol{\varphi} - \boldsymbol{\theta}$ into consideration. Taking the derivative of the $p$-th column of matrix $\boldsymbol{\Phi}_j\boldsymbol{\Psi}_j$ with respect to $\theta_p$, we get

$$\boldsymbol{b}_{jp} = \boldsymbol{j}\frac{2\pi f_c}{c}e^{\boldsymbol{j}\frac{2\pi f_c}{c}\frac{\partial\gamma_{jp}^{\mathrm{r}}}{\partial\theta_p}}\boldsymbol{\Phi}_j\boldsymbol{X}\boldsymbol{u}_p + e^{\boldsymbol{j}\frac{2\pi f_c}{c}\gamma_{jp}^{\mathrm{r}}}\boldsymbol{\Phi}_j\boldsymbol{X}\frac{\partial\boldsymbol{u}_p}{\partial\theta_p}, \tag{2.42}$$

According to (2.19), the $p$-th column of matrix $\boldsymbol{B}$ consists of $\boldsymbol{b}_{jp}$ for $\forall j$, i.e. $\boldsymbol{b}_p = [\boldsymbol{b}_{1p}^{\mathrm{T}}, \ldots, \boldsymbol{b}_{M_{\mathrm{R}}p}^{\mathrm{T}}]^{\mathrm{T}}$. We also have

$$\frac{\partial\boldsymbol{u}_p}{\partial\theta_p} = [\boldsymbol{j}\frac{2\pi f_c}{c}e^{\boldsymbol{j}\frac{2\pi f_c}{c}\frac{\partial\gamma_{1p}^{\mathrm{t}}}{\partial\theta_p}}, \cdots, \boldsymbol{j}\frac{2\pi f_c}{c}e^{-\boldsymbol{j}\frac{2\pi f_c}{c}\frac{\partial\gamma_{M_{\mathrm{T}}p}^{\mathrm{t}}}{\partial\theta_p}}]^{\mathrm{T}}. \tag{2.43}$$

After getting the matrix $\boldsymbol{B}$, (JS) optimization framework in (2.7) can be implemented to detect the targets' angular locations. More details will be explored in the numerical examples.

## 2.4 Fast algorithms

Using interior point methods can be time consuming for large problems. In order to speed up the computing process for (JS) and (BJS) in (2.7), (2.21), we can use a first order method based on a proximal operator, namely the Fast Iterative Shrinkage-Thresholding Algorithm (FISTA) [14]. In this section, we first review the key concept in FISTA. The implementation of FISTA for (JS) is straightforward, while (BJS) requires more effort since it has convex constraints in the optimization problem. A smoothing function [43] is introduced to approximate $\|\boldsymbol{x}\|_{2,1}$ in order to implement FISTA, and continuation techniques [44] based on the smoothing parameter are introduced to further increase the convergence speed.

### 2.4.1 Review: FISTA and proximal operator

To introduce the algorithm, we first review a key concept used in FISTA, named Moreau's proximal operator, or proximal operator for short [45]. For a closed proper convex function $h : \mathbb{R}^N \to \mathbb{R} \cup \{\infty\}$, the proximal operator of $h$ is defined by

$$\text{prox}_h(\boldsymbol{x}) = \underset{\boldsymbol{u}\in\mathbb{R}^N}{\arg\min} \left\{ h(\boldsymbol{u}) + \frac{1}{2}\|\boldsymbol{u} - \boldsymbol{x}\|_2^2 \right\}. \tag{2.44}$$

The proximal operator is a key step in FISTA that solves the following composite nonsmooth problem:

$$\min_{\boldsymbol{x}\in\mathbb{R}^N} F(\boldsymbol{x}) = f(\boldsymbol{x}) + g(\boldsymbol{x}), \tag{2.45}$$

where $f : \mathbb{R}^N \to \mathbb{R}$ is a smooth convex function, and it is continuously differentiable with a Lipschitz continuous gradient $L_{\nabla f}$:

$$\|\nabla f(\boldsymbol{x}) - \nabla f(\boldsymbol{z})\|_2 \leq L_{\nabla f}\|\boldsymbol{x} - \boldsymbol{z}\|_2, \quad \text{for all } \boldsymbol{x}, \boldsymbol{z} \in \mathbb{R}^N, \tag{2.46}$$

and $g : \mathbb{R}^N \to \mathbb{R} \cup \{\infty\}$ is continuous convex function which is possibly nonsmooth. The FISTA algorithm is given as follows.

**Fast Iterative Shrinkage-Thresholding Algorithm**

**Input**: An upper bound $L \geq L_{\nabla f}$.
**Step 0.** Take $\boldsymbol{z}_1 = \boldsymbol{x}_0, t_1 = 1$.
**Step k.** ($k \geq 1$) Compute
$\boldsymbol{x}_k = \text{prox}_{\frac{1}{L}g}\left(\boldsymbol{z}_k - \frac{1}{L}\nabla f(\boldsymbol{z}_k)\right)$.
$t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2}$.
$\boldsymbol{z}_{k+1} = \boldsymbol{x}_k + \frac{t_k - 1}{t_{k+1}}(\boldsymbol{x}_k - \boldsymbol{x}_{k-1})$.

Table 2.1: Fast iterative shrinkage-thresholding algorithm

The convergence rate of the sequence generated by FISTA is determined by the following theorem from [14].

**Theorem 2.3.** *Let $\{\boldsymbol{x}_k\}_{k\geq 0}$ be generated by FISTA, and let $\hat{\boldsymbol{x}}$ be an optimal solution of (4.11). Then for any $k \geq 1$,*

$$F(\boldsymbol{x}_k) - F(\hat{\boldsymbol{x}}) \leq \frac{2L_{\nabla f}\|\boldsymbol{x}_0 - \hat{\boldsymbol{x}}\|_2^2}{(k+1)^2}. \tag{2.47}$$

### 2.4.2 FISTA for compressed sensing with structured dictionary mismatches

For optimization framework (JS), we know that $f(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{\Phi}\boldsymbol{x} - \boldsymbol{y}\|_2^2$, then the Lipschitz constant is equal to $\|\boldsymbol{\Phi}\|_2^2$. When $g(\boldsymbol{x}) = \lambda\|\boldsymbol{x}\|_{2,1}$ and $\boldsymbol{x} \in \mathbb{R}^{2N}$, the proximal operator of $\boldsymbol{x} = [\boldsymbol{s}^{\mathrm{T}}, \boldsymbol{p}^{\mathrm{T}}]^{\mathrm{T}}$ is a group-thresholding operator defined as

$$\text{prox}_{\alpha g}(\{[x_i, x_{i+N}]\}) = \frac{[x_i, x_{i+N}]}{\sqrt{x_i^2 + x_{i+N}^2}}\max(\sqrt{x_i^2 + x_{i+N}^2} - \alpha\lambda, 0), \quad 1 \leq i \leq N. \tag{2.48}$$

Please note that this proximal operator yield $[0, 0]$ when $x_i = x_{i+N} = 0$. Hence, the algorithm using FISTA for (JS) is straightforward and summarized as follows:

**FISTA for Joint Sparse Recovery**

**Input**: An upper bound $L \geq \|\boldsymbol{\Phi}\|_2^2$ and initial point $\boldsymbol{x}_0$.
**Step 0.** Take $\boldsymbol{z}_1 = \boldsymbol{x}_0, t_1 = 1$.
**Step k.** ($k \geq 1$) Compute
$\nabla f(\boldsymbol{z}_k) = \boldsymbol{\Phi}^{\mathrm{T}}(\boldsymbol{\Phi}\boldsymbol{z}_k - \boldsymbol{y})$,
$\boldsymbol{x}_k = \mathrm{prox}_{\frac{1}{L}g}\left(\boldsymbol{z}_k - \frac{1}{L}\nabla f(\boldsymbol{z}_k)\right)$, and $g(\boldsymbol{u}) = \lambda\|\boldsymbol{u}\|_{2,1}$,
$t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2}$,
$\boldsymbol{z}_{k+1} = \boldsymbol{x}_k + \frac{t_k - 1}{t_{k+1}}(\boldsymbol{x}_k - \boldsymbol{x}_{k-1})$.

Table 2.2: FISTA for joint sparse recovery

The FISTA implementation of (BJS) needs more work due to the positive and bounded constraints in the optimization. In order to use FISTA, we write these two convex constraints as an indicator function in the objective function. Then (BJS) is transformed into

$$\begin{aligned} \min_{\boldsymbol{s},\boldsymbol{p},\boldsymbol{x}} \quad & \tfrac{1}{2}\|\boldsymbol{A}\boldsymbol{s} + \boldsymbol{B}\boldsymbol{p} - \boldsymbol{y}\|_2^2 + \lambda\|\boldsymbol{x}\|_{2,1} + I_{\mathcal{F}}(\boldsymbol{s},\boldsymbol{p}), \\ \text{s.t.} \quad & \boldsymbol{x} = [\boldsymbol{s}^{\mathrm{T}}, \boldsymbol{p}^{\mathrm{T}}]^{\mathrm{T}}, \end{aligned} \tag{2.49}$$

where $I_{\mathcal{F}}(\boldsymbol{s},\boldsymbol{p})$ is the indicator function for set $\mathcal{F} = \{\boldsymbol{s} \geq 0, -r\boldsymbol{s} \leq \boldsymbol{p} \leq r\boldsymbol{s}\}$. FISTA cannot be implemented directly since there are two nonsmooth functions, i.e., $\|\boldsymbol{x}\|_{2,1}$ and $I_{\mathcal{F}}(\boldsymbol{s},\boldsymbol{p})$, in the objective function.

One way to solve this issue is to approximate $h(\boldsymbol{x}) = \lambda\|\boldsymbol{x}\|_{2,1}$ by its Moreau envelope [45], given as

$$h_\mu(\boldsymbol{x}) = \min_{\boldsymbol{u}\in\mathbb{R}^{2N}} \left\{ h(\boldsymbol{u}) + \frac{1}{2\mu}\|\boldsymbol{u} - \boldsymbol{x}\|_2^2 \right\}. \tag{2.50}$$

The Moreau envelope $h_\mu$ is continuously differentiable, and its gradient is equal to

$$\nabla h_\mu(\boldsymbol{x}) = \frac{1}{\mu}(\boldsymbol{x} - \mathrm{prox}_{\mu h}(\boldsymbol{x})), \tag{2.51}$$

which is Lipschitz continuous with constant $1/\mu$ and can be computed using (2.48). The smoothing approximation is more accurate with smaller $\mu$. For more details, please check [43].

By letting $f(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{\Phi x} - \boldsymbol{y}\|_2^2$ and $g(\boldsymbol{x}) = I_{\mathcal{F}}(\boldsymbol{s}, \boldsymbol{p})$, the smoothed (BJS) can be presented as

$$(\mu\text{BJS}) \quad \min_{\boldsymbol{x}} \quad f(\boldsymbol{x}) + h_\mu(\boldsymbol{x}) + g(\boldsymbol{x}). \tag{2.52}$$

The Lipschitz constant for the gradient of $f(\boldsymbol{x}) + h_\mu(\boldsymbol{x})$ is $\|\boldsymbol{\Phi}\|_2^2 + \frac{1}{\mu}$. In order to implement FISTA, the proximal operator of $g(\boldsymbol{x})$ is needed and can be expressed as a projection onto the set $\mathcal{F}$:

$$\text{prox}_g(\boldsymbol{x}) = P_{\mathcal{F}}([\boldsymbol{s}^{\mathrm{T}}, \boldsymbol{p}^{\mathrm{T}}]^{\mathrm{T}}). \tag{2.53}$$

Since the convex set $\mathcal{F}$ can be expressed as $\mathcal{F} = \bigcap_{i=1}^{N} \mathcal{F}_i$, where $\mathcal{F}_i = \{s_i \geq 0, -rs_i \leq p_i \leq rs_i\}$, the proximal operator can be computed element-wise, i.e.,

$$\text{prox}_g(s_i, p_i) = P_{\mathcal{F}_i}(s_i, p_i). \tag{2.54}$$

Here the projection from $[s_i, p_i]$ onto the two dimensional convex cone $\mathcal{F}_i$ is easy and given as follows,

$$P_{\mathcal{F}_i}(s_i, p_i) = \begin{cases} (s_i, p_i) & -rs_i \leq p_i \leq rs_i, \\ (0,0) & \frac{s_i}{r} \leq p_i \leq -\frac{s_i}{r}, \\ c(1, r) & rs_i \leq p_i, -\frac{s_i}{r} \leq p_i, \\ c(1, -r) & -rs_i \geq p_i, \frac{s_i}{r} \geq p_i, \end{cases} \tag{2.55}$$

where $c = \frac{s_i + |rp_i|}{1+r^2}$. Hence the FISTA implementation for ($\mu$BJS) is given in the table 2.3.

As we discussed earlier, smaller $\mu$ leads to better approximation accuracy. However, smaller $\mu$ incurs a larger $L$ in the algorithm, which forces the algorithm running longer to converge. The continuation technique was utilized in [44, 15] to resolve this issue. The idea of continuation is to solve ($\mu$BJS) with $\mu_1 \geq \mu_2 \geq \ldots \geq \mu_f$ sequentially, and use the previous solution to warm start the next optimization.

**FISTA for $\mu$-Smoothed (BJS) Recovery**

**Input**:
An upper bound $L \geq \|\boldsymbol{\Phi}\|_2^2 + \frac{1}{\mu}$ and initial point $\boldsymbol{x}_0$.
**Step 0.** Take $\boldsymbol{z}_1 = \boldsymbol{x}_0, t_1 = 1$.
**Step k.** ($k \geq 1$) Compute
$\nabla f(\boldsymbol{z}_k) = \boldsymbol{\Phi}^{\mathrm{T}}(\boldsymbol{\Phi}\boldsymbol{z}_k - \boldsymbol{y})$,
$\nabla h_\mu(\boldsymbol{z}_k) = \frac{1}{\mu}(\boldsymbol{z}_k - \mathrm{prox}_{\mu h}(\boldsymbol{z}_k))$,
$\boldsymbol{x}_k = P_{\mathcal{F}}\left(\boldsymbol{z}_k - \frac{1}{L}\nabla f(\boldsymbol{z}_k) - \frac{1}{L}\nabla h_\mu(\boldsymbol{z}_k)\right)$,
$t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2}$,
$\boldsymbol{z}_{k+1} = \boldsymbol{x}_k + \frac{t_k - 1}{t_{k+1}}(\boldsymbol{x}_k - \boldsymbol{x}_{k-1})$.

Table 2.3: FISTA for $\mu$-smoothed (BJS) recovery

## 2.5 Numerical examples

In this section, we present several numerical examples to show the advantages of using the joint sparse recovery method when dictionary mismatches exist in compressed sensing. In the first example, we randomly generate the data and mismatch parameters following Gaussian distributions. The measurement are obtained using model (2.2). FISTA-based joint sparse method and the alternating minimization method [21] are considered in this case. We show that the joint sparse method provides a better reconstruction with less computational effort. In the last two examples, we compare the joint sparse method with P-BPDN [20] under both passive and active sensing scenarios. Please note that P-BPDN is also equivalent to the reconstruction method proposed in [30].

### 2.5.1 Randomly generated data

In this numerical example we compare the FISTA-based joint-sparse method with the alternating minimization method proposed in [21] when they are applied in the optimization (2.2). Both matrices $\boldsymbol{A} \in \mathbb{R}^{M\times N}$ and $\boldsymbol{B} \in \mathbb{R}^{M\times N}$ are randomly generated with a normal

distribution with mean 0 and standard deviation 1. We set $N = 100$. The noise term $\boldsymbol{w}$ is randomly generated according to a normal distribution with mean zero and standard deviation $\sigma_n = 0.1$. The mismatch term $\boldsymbol{\beta}$ is also generated according to a normal distribution with standard deviation $\delta = 1$. $\lambda$ is chosen as $10\sigma_n\sqrt{2\log(N)}$.

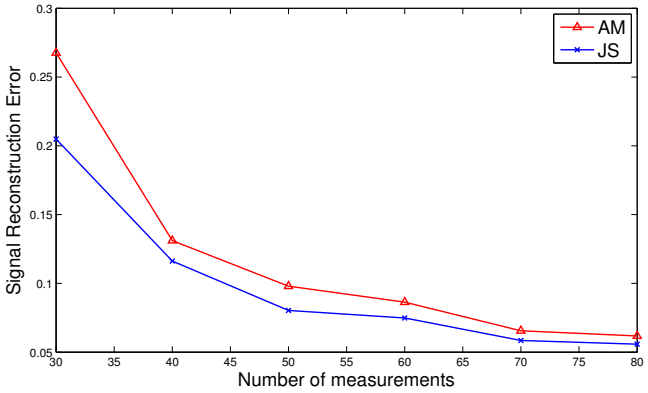


Figure 2.1: Signal reconstruction error with different number of measurements.

In the first comparison, we range the number of measurements $M$ from 30 to 80. The sparsity of the signal $\boldsymbol{s}$ is 3. We use $\|\boldsymbol{s} - \hat{\boldsymbol{s}}\|_2/\|\boldsymbol{s}\|_2$ to denote the signal reconstruction error. We run 50 Monte Carlo iterations at each testing point. We can see from Fig. 2.1 that (JS) with FISTA performs uniformly better than the alternating minimization method. The average CPU time for alternating minimization is 15.61s, while (JS) needs only 0.26s.

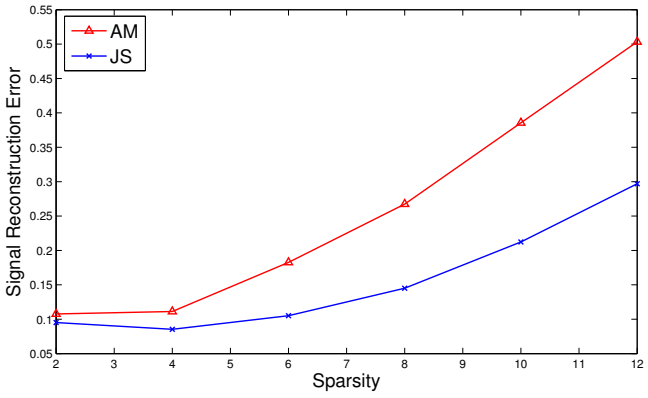


Figure 2.2: Signal reconstruction error with different sparsity level.

Next, we range the sparsity level $K$ from 2 to 12 to compare these two methods. The number of measurements is 50. From Fig. 2.2, we can see that (JS) has a uniformly smaller

reconstruction error. The average CPU time for (JS) is 0.42s, while the CPU time for alternating minimization is 14.34s.

### 2.5.2 Nonuniform linear array using off-grid compressed sensing

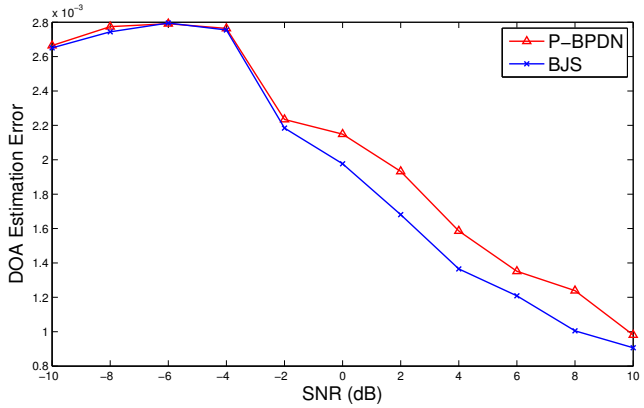


Figure 2.3: DOA estimation error with different SNR ($T = 1000$).

In this subsection, we consider a passive sensing simulation with a nonuniform linear array. The array for this part consists of two subarrays. One has sensors located at $id$ with $1 \leq i \leq 5$ while the other has sensors located at $6jd$ with $1 \leq j \leq 6$, and $d$ is half of the wavelength. This configuration is also called a nested array, as proposed in [41]. We compare the optimization formulation (BJS) with P-BPDN in this experiment. The power of the noise is assumed to be known; if not, an estimation of it can be easily incorporated into the (BJS) formulation. The area we are interested ranges from $\sin(\theta) = -1$ to $\sin(\theta) = 1$, with a step size of 0.01. We randomly generate 15 targets with the same signal power. The noise at each sensor is randomly generated as white Gaussian noise with power $\sigma_n^2$. $\lambda$ in the LASSO formulation is chosen to be $\sigma_n\sqrt{2\log(N)}$ according to [12]. However, since we use only first-order Taylor expansion to approximate the system matrix $\boldsymbol{A}(\boldsymbol{\theta})$, the scale of the error is far larger than the additive Gaussian noise. Therefore we chose $\lambda = 20\sigma_n\sqrt{2\log(N)}$ in our simulation. Here $N$ is the dimension of the signal of interest.

First we range the signal to noise ratio (SNR) from $-10$ dB to 10 dB in Fig. 2.3. The number of time samples used to estimate (2.24) is $T = 1000$. In Fig. 2.4, we range $T$, with

the SNR fixed at 0 dB. The DOA error is computed with respect to $\sin(\theta)$. Both figures show that (BJS) yields better DOA estimation accuracy than P-BPDN.

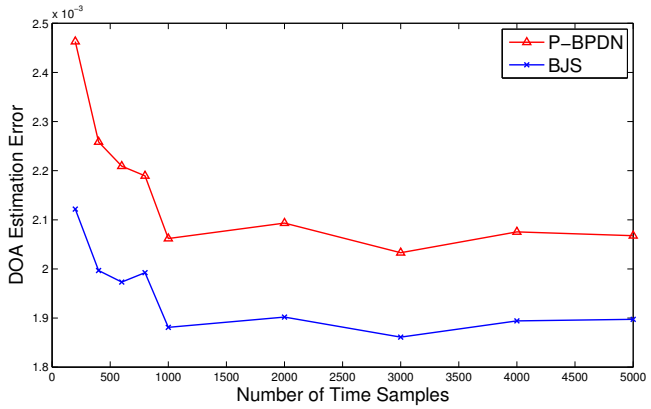


Figure 2.4: DOA estimation error with different T (SNR= 0 dB).

The interior method for (BJS) works well when the dimension of the problem is small. In the next simulation, we increase the number of sensors in the linear array. The array consists of two subarrays. One has sensors located at $id$ with $1 \leq i \leq 10$ while the other has sensors located at $11jd$, with $1 \leq j \leq 12$. We randomly generate 26 targets with the same signal power. We run the ($\mu$BJS) using FISTA with a continuation scheme. Let $\mu_f = 10^{-8}\lambda^{-1}$. The DOA estimation results are shown in Fig. 2.5. The running time for ($\mu$BJS) with FISTA is 4.92s, while (BJS) with the interior point method takes 63.09s. They both have a DOA estimation error of $5.5 \times 10^{-4}$.

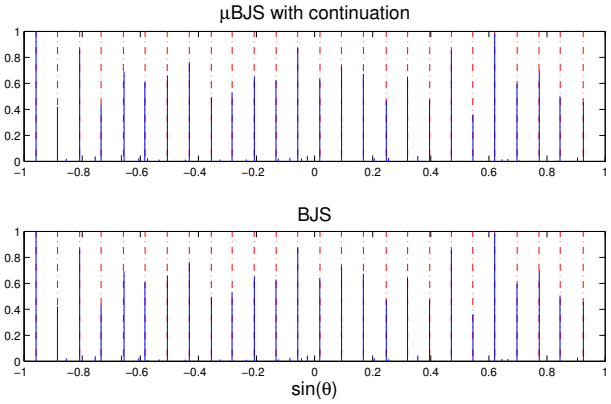


Figure 2.5: Normalized spectrum for ($\mu$BJS) with continuation, and (BJS) (T=500, SNR= 10 dB).

### 2.5.3 MIMO radar using off-grid compressed sensing

In this numerical example, we compare FISTA based (JS) with P-BPDN [20] in a MIMO radar scenario. To fully explore the diversity of the model, we consider a MIMO system with 30 transmitters and 10 receivers whose locations are randomly generated within a disk with a radius of 5 meters. The carrier frequency $f_c$ is 1 GHz. Each transmitter sends out uncorrelated QPSK waveforms. The signal to noise ratio (SNR) is defined to be the ratio of the power of the transmitted waveform to the power of the additive noise in the receivers. We are interested in the area ranging from $-40°$ to $40°$, with step size $1°$. We assume that two targets are at least $1°$ apart. We take $L = 50$ samples for each receiver and then compress the received signal to dimension $M = 10$. Therefore we chose $\lambda = 50\sigma_n\sqrt{2\log(N)}$ in our simulation.

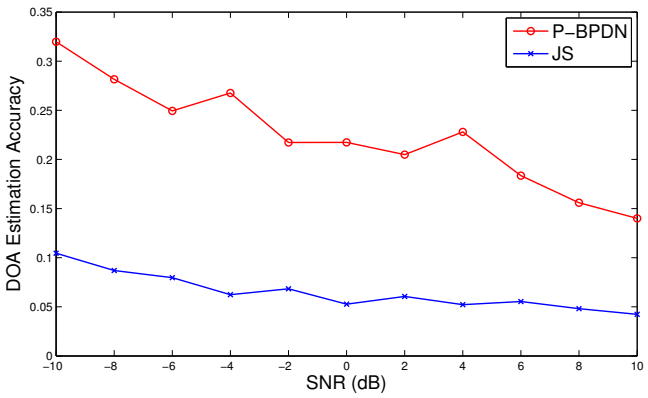


Figure 2.6: DOA estimation performance for two closely located targets with a MIMO radar system.

In the first simulation, we compare these two algorithms with two closely-spaced targets with SNR ranging from $-10$dB to 10dB and show how joint sparsity benefits the reconstruction. The locations of the two targets are randomly generated from the intervals $[16.5°, 17.5°]$ and $[18.5°, 19.5°]$, with equal signal power. We run 50 Monte Carlo iterations of every value of SNR, with the results shown in Fig. 2.6. The DOA estimation error in the figure is the average DOA estimation error in degrees. We can see that the method proposed in this chapter has consistent better reconstruction performance than P-BPDN for location estimation.

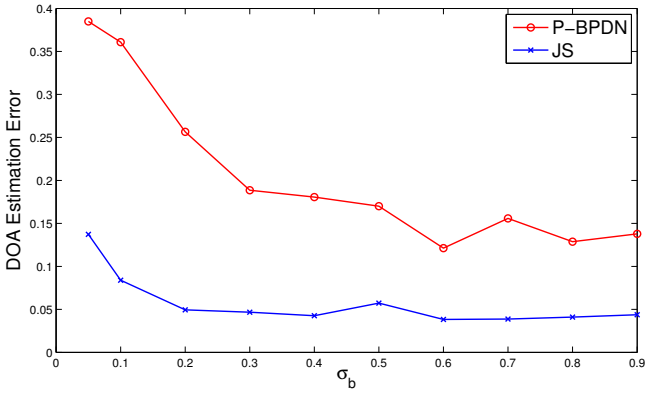


Figure 2.7: DOA estimation error with changing $\sigma_b$ ($\sigma_a = 1$).

In the next simulation, we compare (JS) using FISTA with P-BPDN when the dynamic range changes between these two targets. Suppose the first target is randomly generated with signal power $\sigma_a^2 = 1$, and the second target has a signal power $\sigma_b^2$. SNR is chosen to be 10 dB in this case. From Fig. 2.7 we can see that (JS) performs better with respect to changing dynamic range.

## 2.6 Summary

In this chapter we considered structured dictionary mismatches in sparse recovery. We proposed to use the joint sparse recovery model to exploit the relation between mismatch parameters and the original sparse signal. A performance bound on the joint sparse reconstruction was derived. For off-grid DOA estimations, a bounded joint sparse recovery method was implemented. However, solving this optimization using the interior point method is highly inefficient, thus fast algorithms based on FISTA were implemented to yield efficient outcomes. We demonstrated the usefulness of the proposed algorithms in the applications of MIMO radar and nested arrays. Numerical examples were presented to compare the performance of the joint sparse method with other methods, such as alternating minimization. We demonstrated that by exploiting the joint sparse property, we can get more accurate reconstruction results than the previous methods with less computational time.

# Chapter 3

# Continuous Sparse Recovery For DOA Estimation with Co-prime Arrays

In this chapter we will extend the off-grid model from the previous chapter into a continuous domain. In particular, we consider a recently proposed form of passive arrays, co-prime arrays, to increase the degrees of freedom.[2]

## 3.1 Introduction

In the last few decades, research on direction of arrival (DOA) estimation using array processing has focused primarily on uniform linear arrays (ULA) [46]. It is well known that using a ULA with $N$ sensors, the number of sources that can be resolved by MUSIC-like algorithms is $N-1$ [47]. New geometries [41, 42] of non-uniform linear arrays have been recently proposed to increase the degrees of freedom of the array by exploiting the covariance matrix of the received signals. Vectorizing the covariance matrix, the system model can be viewed as a virtual array with a wider aperture. In [41], a nested array structure was proposed to increase the degrees of freedom from $O(N)$ to $O(N^2)$, with only $O(N)$ sensors. However, some of the sensors in this structure are closely located, which leads to mutual coupling among these sensors. To overcome this shortcoming, co-prime arrays were proposed in [42]. Such arrays consist of two subarrays with $M$ and $N$ sensors respectively.

[2]This chapter is based on Z. Tan, Y. C. Eldar and A. Nehorai,"Direction of arrival estimation with co-prime arrays: a super resolution viewpoint," *IEEE Trans. Signal Processing*, Vol. 62, pp. 5565-5576, Nov. 2014. 

It was shown that by using $O(M+N)$ number of sensors, this structure can achieve $O(MN)$ degrees of freedom. In this chapter we focus on co-prime arrays.

The increased degrees of freedom provided by the co-prime structure can be utilized to improve DOA estimation. To this end, two main methodologies have been proposed to utilize this increased degrees of freedom for co-prime arrays. The first are subspace methods, such as the MUSIC algorithm. In [48], a spatial smoothing technique was implemented prior to the application of MUSIC, and the authors showed that an increased number of sources can be detected. However, the application of spatial smoothing reduces the obtained virtual array aperture [49]. The second approach uses sparsity-based recovery to overcome these disadvantages of subspace methods [49]-[50]. Traditional sparsity techniques discretize the range of interest onto a grid. Off-grid targets can lead to mismatches in the model and deteriorate the performance significantly [18]. In [51, 20] the grid mismatches are estimated simultaneously with the original signal, leading to improved performance over traditional sparse recovery methods. In [52], the joint sparsity between the original signal and the mismatch is exploited during DOA estimation. Due to the first-order approximation used in [52], the estimation performance is still limited by higher-order modeling mismatches.

To overcome grid mismatch of traditional sparsity-based methods, in this chapter we apply the recently developed mathematical theory of continuous sparse recovery for super resolution [22]-[54] to DOA estimation with co-prime arrays. The term "super resolution" in this chapter is related to the off-grid problem and is different from the traditional definition commonly used in DOA estimation. In [22][53] it was shown that assuming a signal consists of spikes, the high frequency content of the signal's spectrum can be perfectly recovered in a robust fashion by sampling only the low end of its spectrum, when the minimum distance between spikes satisfies certain requirements. In [54], the author provides performance guarantees on the recovered support set of the sparse signal. One merit of this theory is that it considers all possible locations within the desired range, and thus does not suffer from model mismatches.

Here we extend the mathematical theory of super resolution to DOA estimation with co-prime arrays under Gaussian noise. The effective noise resulting from the usage of co-prime arrays consists of a term with a known structure and another term containing quadratic combinations of Gaussian noise. Therefore, we modify the reconstruction method to fit

these particular noise properties and prove the robustness of our approach by analyzing the noise statistics. We also prove that with $2M + N$ sensors in a co-prime array, it is possible to detect up to $O(MN)$ sources robustly. Previous identifiability research using traditional compressed seining for co-prime arrays [50] was based on the idea of mutual coherence [55]. Using mutual coherence it can be shown that co-prime arrays increase the number of detected sources from $O(M + N)$ to $O(MN)$, but this analysis is valid only for very small values of the number of sources.

Source number detection is another main application of array processing. Various methods have been proposed over the years based on the eigenvalues of the signal space, such as the Akaike information criterion [56], second-order statistic of eigenvalues (SORTE) [57], and the predicted eigen-threshold approach [58]. The authors of [59] showed that among these methods, SORTE often leads to better detection performance. Here we combine the SORTE approach with the spectrum reconstructed from the proposed DOA estimation algorithm to determine the number of sources. Through this source number detection, we identify which reconstructed spikes are true detections and which are false alarms.

The chapter is organized as follows. In Section 3.2, we introduce the DOA estimation model and explain how co-prime arrays can increase the degrees of freedom. In Section 3.3, we adapt the theory of super resolution to co-prime arrays, and analyze the robustness of this extension by studying the statistics of the noise pattern in the model. We propose a numerical method to perform DOA estimation for co-prime arrays in Section 3.4 and we then extend this approach to detect the number of sources. Section 3.5 presents extensive numerical simulations demonstrating the advantages of our method in terms of estimation accuracy, degrees of freedom, and resolution ability.

## 3.2 DOA estimation and co-prime arrays

Consider a linear sensor array with $L$ sensors which may be non-uniformly located. Assume that there are $K$ narrow band sources located at $\theta_1, \theta_2, \ldots, \theta_K$ with signal powers $\sigma_1^2, \sigma_2^2, \ldots, \sigma_K^2$. The steering vector for the $k$th source located at $\theta_k$ is $\boldsymbol{a}(\theta_k) \in \mathbb{C}^{L\times 1}$ with $l$th element $e^{\boldsymbol{\jmath}(2\pi/\lambda)d_l \sin(\theta_k)}$, in which $d_l$ is the location of the $l$th sensor and $\lambda$ is the wavelength. The data collected by all sensors at time $t$ can be expressed as

$$\boldsymbol{x}(t) = \sum_{k=1}^{K} \boldsymbol{a}(\theta_k) s_k(t) + \boldsymbol{\varepsilon}(t) = \boldsymbol{A}\boldsymbol{s}(t) + \boldsymbol{\varepsilon}(t), \tag{3.1}$$

for $t = 1, \ldots, T$, where $\boldsymbol{\varepsilon}(t) = [\varepsilon_1(t), \varepsilon_2(t), \ldots, \varepsilon_L(t)]^{\mathrm{T}} \in \mathbb{C}^{L\times 1}$ is an i.i.d. white Gaussian noise $\mathcal{CN}(0, \sigma^2)$, $\boldsymbol{A} = [\boldsymbol{a}(\theta_1), \boldsymbol{a}(\theta_2) \ldots, \boldsymbol{a}(\theta_K)] \in \mathbb{C}^{L\times K}$, and $\boldsymbol{s}(t) = [s_1(t), s_2(t), \ldots, s_K(t)]^{\mathrm{T}}$ represents the source signal vector with $s_k(t)$ distributed as $\mathcal{CN}(0, \sigma_k^2)$. We assume that the sources are temporally uncorrelated.

The correlation matrix of the data can be expressed as

$$\begin{aligned} \boldsymbol{R}_{xx} =& E[\boldsymbol{x}(t)\boldsymbol{x}^*(t)] \\ =& \boldsymbol{A}\boldsymbol{R}_{ss}\boldsymbol{A}^* + \sigma^2\boldsymbol{I} \\ =& \sum_{k=1}^{K} \sigma_k^2 \boldsymbol{a}(\theta_k)\boldsymbol{a}^*(\theta_k) + \sigma^2\boldsymbol{I}, \end{aligned} \tag{3.2}$$

in which $\boldsymbol{R}_{ss}$ is a $K \times K$ diagonal matrix with diagonal elements $\sigma_1^2, \sigma_2^2, \ldots, \sigma_K^2$. After vectorizing the correlation matrix $\boldsymbol{R}_{xx}$, we have

$$\boldsymbol{z} = \mathrm{vec}(\boldsymbol{R}_{\boldsymbol{xx}}) = \boldsymbol{\Phi}(\theta_1, \theta_2, \ldots, \theta_K)\boldsymbol{s} + \sigma^2\mathbf{1}_n, \tag{3.3}$$

where

$$\boldsymbol{\Phi}(\theta_1, \ldots, \theta_K) = \boldsymbol{A}^* \odot \boldsymbol{A} = [\boldsymbol{a}(\theta_1)^{\mathrm{H}} \otimes \boldsymbol{a}(\theta_1), \ldots, \boldsymbol{a}(\theta_K)^{\mathrm{H}} \otimes \boldsymbol{a}(\theta_K)], \tag{3.4}$$

$\boldsymbol{s} = [\sigma_1^2, \sigma_2^2, \ldots, \sigma_K^2]^{\mathrm{T}}$, and $\mathbf{1}_n = [\boldsymbol{e}_1^{\mathrm{T}}, \boldsymbol{e}_2^{\mathrm{T}}, \ldots, \boldsymbol{e}_L^{\mathrm{T}}]^{\mathrm{T}}$ with $\boldsymbol{e}_i$ denoting a vector with all zero elements, except for the $i$th element, which equals one.

Comparing (3.1) with (3.3), we see that $\boldsymbol{s}$ behaves like a coherent source and $\sigma^2\mathbf{1}_n$ becomes a deterministic noise term. The distinct rows in $\boldsymbol{\Phi}$ act as a larger virtual array with sensors located at $d_i - d_j$, with $1 \le i, j \le L$. Traditional DOA estimation algorithms can be implemented to detect more sources when the structure of the sensor array is properly designed. Following this idea, nested arrays [41] and co-prime arrays [42] were introduced, and then shown to increase the degrees of freedom from $O(N)$ to $O(N^2)$, and from $O(M+N)$

to $O(MN)$ respectively. In the following, we focus only on co-prime arrays; the results follow naturally for nested arrays.

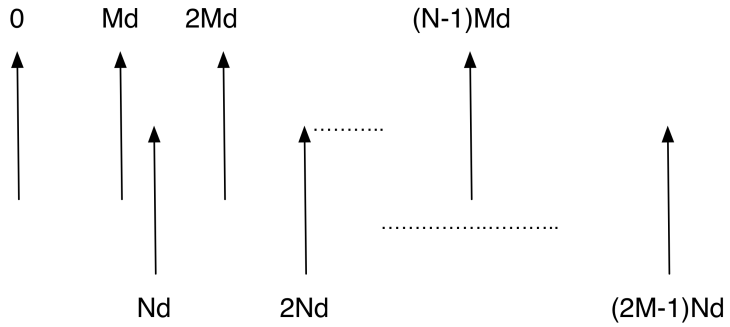


Figure 3.1: Geometry of co-prime arrays.

Consider a co-prime array structure consisting of two arrays with $N$ and $2M$ sensors respectively. The locations of the $N$ sensors are in the set $\{Mnd, 0 \leq n \leq N-1\}$, and the locations of the $2M$ sensors are in the set $\{Nmd, 0 \leq m \leq 2M-1\}$ as illustrated in Fig. 3.1. The first sensors of these two arrays are collocated. The geometry of such a co-prime array is shown in Fig. 3.1. The locations of the virtual sensors in $\mathbf{\Phi}$ from (3.3) are given by the cross difference set $\{\pm(Mn-Nm)d, 0 \leq n \leq N-1, 0 \leq m \leq 2M-1\}$ and the two self difference sets. In order to implement spatial smoothing of MUSIC, or to use other popular DOA estimation techniques, we are interested in generating a consecutive range of virtual sensors. It was shown in [48] that when $M$ and $N$ are co-prime numbers, a consecutive range can be created from $-MNd$ to $MNd$, with $\{-MNd, -(MN-1)d, \ldots, -2d, -d, d, 2d, \ldots, (MN-1)d, MNd\}$ taken from the cross difference set and $\{0d\}$ taken from any one of the self difference sets.

By removing repeated rows of (3.3) and sorting the remaining rows from $-MNd$ to $MNd$, we have the linear model rearranged as

$$\tilde{\boldsymbol{z}} = \tilde{\mathbf{\Phi}}\boldsymbol{s} + \sigma^2\tilde{\boldsymbol{w}}. \tag{3.5}$$

It is easy to verify that $\tilde{\boldsymbol{w}} \in \mathbb{R}^{(2MN+1)\times 1}$ is a vector whose elements all equal zero, except the $(MN+1)$th element, which equals one. The matrix $\tilde{\boldsymbol{\Phi}} \in \mathbb{R}^{(2MN+1)\times K}$ is given by

$$\tilde{\boldsymbol{\Phi}} = \begin{bmatrix} e^{-\boldsymbol{j}MNd\frac{2\pi}{\lambda}\sin(\theta_1)} & \dots & e^{-\boldsymbol{j}MNd\frac{2\pi}{\lambda}\sin(\theta_K)} \\ e^{-\boldsymbol{j}(MN-1)d\frac{2\pi}{\lambda}\sin(\theta_1)} & \dots & e^{-\boldsymbol{j}(MN-1)d\frac{2\pi}{\lambda}\sin(\theta_K)} \\ \vdots & \ddots & \vdots \\ e^{\boldsymbol{j}MNd\frac{2\pi}{\lambda}\sin(\theta_1)} & \dots & e^{\boldsymbol{j}MNd\frac{2\pi}{\lambda}\sin(\theta_K)} \end{bmatrix},$$

which is the steering matrix of a ULA with $2MN+1$ sensors. Therefore, (3.5) can be regarded as a ULA detecting a coherent source $\boldsymbol{s}$ with deterministic noise term $\sigma^2\tilde{\boldsymbol{w}}$. By applying MUSIC with spatial smoothing, the authors in [48] showed that $O(MN)$ sources can be detected, using this approach.

## 3.3 Co-prime arrays with super resolution

In this section we first assume that the signal model (3.3) is accurate, which means that the number of samples $T$ is infinite, and also that the noise power $\sigma^2$ is known a priori. The super resolution theory developed in [22] can then be applied to co-prime arrays to demonstrate that we can detect up to $O(MN)$ sources robustly as long as the distance between any two sources is on the order of $\frac{1}{MN}$. We then consider the case in which the number of time samples $T$ is limited and demonstrate the robustness of super resolution recovery via statistical analysis of the noise structure.

### 3.3.1 Mathematical theory of super resolution

Super resolution seeks to recover high frequency details from the measurement of low frequency components. Mathematically, given a measure $s(\tau)$ with $\tau \in [0,1]$, the Fourier series coefficients are recorded as

$$r(n) = \int_0^1 e^{-\boldsymbol{j}2\pi n\tau} s(\tau)d\tau, \quad n = -f_c, -f_c+1, \dots, f_c. \tag{3.6}$$

Using the operator $\boldsymbol{F}$ to denote the low frequency measuring operator which transforms a signal from its continuous time domain into its discrete frequency domain, we can represent (3.6) as $\boldsymbol{r} = \boldsymbol{F}s$, in which $\boldsymbol{r} = [r(-f_c), \ldots, r(f_c)]^{\mathrm{T}}$ and $s = s(\tau), 0 \leq \tau \leq 1$.

Suppose that the measure $s(\tau)$ is sparse, i.e., $s(\tau)$ is a weighted sum of several spikes:

$$s(\tau) = \sum_{k=1}^{K} s_k \delta_{\tau_k}, \tag{3.7}$$

in which $s_k$ can be complex valued and $\tau_k \in [0, 1]$ for all $k$. Then

$$r(n) = \sum_{k=1}^{K} s_k e^{-\mathrm{j}2\pi n \tau_k}, \quad n = -f_c, -f_c + 1, \ldots, f_c. \tag{3.8}$$

In order to recover $s(\tau)$ from the measurements $r(n)$, total variation minimization is introduced. This criterion encourages the sparsity in the measure $s(\tau)$, just as $\ell_1$ norm minimization produces sparse signals in the discrete space. In the rest of the chapter, we will use $s$ to denote the measure $s(\tau)$ for simplicity. The total variation for the complex measure $s$ is defined as

$$\|s\|_{\mathrm{TV}} = \sup \sum_{j=1}^{\infty} |s(B_j)|, \tag{3.9}$$

the supremum being taken over all partitions of the set $[0, 1]$ into countable collections of disjoint measurable sets $B_j$. When $s$ has the form (3.7), $\|s\|_{\mathrm{TV}} = \sum_{k=1}^{K} s_k$, which resembles the discrete $\ell_1$ norm.

The following convex optimization formula was proposed in [22] to solve the super resolution problem which recovers a sparse measure from $\boldsymbol{r}$:

$$\min_{\tilde{s}} \|\tilde{s}\|_{\mathrm{TV}} \quad \text{s.t.} \quad \boldsymbol{F}\tilde{s} = \boldsymbol{r}. \tag{3.10}$$

When the distance between any two $\tau_i$ and $\tau_j$ is larger than $2/f_c$, then the original sparse signal $s$ is the unique solution to the above convex optimization [22]. The continuous optimization (3.10) can be solved via the dual problem [22]:

$$\begin{aligned} \max_{\boldsymbol{u},\boldsymbol{Q}} \quad & \mathrm{Re}[\boldsymbol{u}^*\boldsymbol{r}] \\ \text{s.t.} \quad & \begin{bmatrix} \boldsymbol{Q} & \boldsymbol{u} \\ \boldsymbol{u}^* & 1 \end{bmatrix} \succeq 0, \\ & \sum_{i=1}^{2MN+1-j} \boldsymbol{Q}_{i,i+j} = \begin{cases} 1 & j = 0, \\ 0 & j = 1, 2, \ldots, 2MN, \end{cases} \end{aligned} \tag{3.11}$$

where $\boldsymbol{Q} \in \mathbb{C}^{(2MN+1)\times(2MN+1)}$ is a Hermitian matrix and $\boldsymbol{u} \in \mathbb{C}^{2MN+1}$ is the Lagrangian multiplier for the constraint $\boldsymbol{F}\tilde{s} = \boldsymbol{r}$. The primal solution $s$ is obtained through a combined process of rooting finding and least-squares [22].

### 3.3.2 DOA estimation with TV-norm minimization

DOA estimation with co-prime arrays can be related to (3.8) by a straightforward change of variables. Letting $\tau_k = \frac{d}{\lambda}(1 - \sin(\theta_k))$ for all $k$, the linear model of (3.5) can be transformed into

$$\begin{aligned} r(n) =& e^{-\boldsymbol{\jmath} 2\pi n \frac{d}{\lambda}}(\tilde{z}_n - \sigma^2 \tilde{w}_n) = e^{-\boldsymbol{\jmath} 2\pi n \frac{d}{\lambda}} \sum_{k=1}^{K} s_k e^{\boldsymbol{\jmath} 2\pi n \frac{d}{\lambda} \sin(\theta_k)} \\ =& \sum_{k=1}^{K} s_k e^{-\boldsymbol{\jmath} 2\pi n \tau_k} = \int_0^1 e^{-\boldsymbol{\jmath} 2\pi n \tau} s(\tau) d\tau, \end{aligned} \tag{3.12}$$

where $n = -MN, -MN+1, \ldots, MN-1, MN$, and $s$ is a sparse measure given in (3.7) with $s_k = \sigma_k^2$. Note that the measure $s$ is different from the vector representation $\boldsymbol{s} = [s_1, \ldots, s_K]^{\mathrm{T}}$, and they are related by (3.7). The change of variables is performed to guarantee that $0 \le \tau_k \le 1$. We use $\mathcal{T} = \{\tau_k, 1 \le k \le K\}$ to denote the support set.

A theorem about the resolution and degrees of freedom for co-prime arrays can be directly derived using Theorem 1.2 in [22]. Before introducing the theorem, we first define the

minimum distance between any two sources as

$$\Delta(\boldsymbol{\theta}) = \min_{\theta_i,\theta_j,\theta_i\neq\theta_j} |\sin(\theta_i) - \sin(\theta_j)|. \tag{3.13}$$

**Theorem 3.1.** *Consider a co-prime array consisting of two linear arrays with $N$ and $2M$ sensors respectively. The distances between two consecutive sensors are $Md$ for the first array and $Nd$ for the second array, where $M$ and $N$ are co-prime numbers, and $d \leq \frac{\lambda}{2}$. Suppose we have $K$ sources located at $\theta_1, \ldots, \theta_K$. If the minimum distance follows the constraint that*

$$\Delta(\boldsymbol{\theta}) \geq \frac{2\lambda}{MNd},$$

*then by solving the convex optimization* (3.10) *with the signal model $\boldsymbol{r} = \boldsymbol{F}s$, one can recover the locations $\theta_k$ for $k = 1, \ldots, K$ exactly. The number of sources that can be detected is*

$$K_{\max} = \frac{MNd}{\lambda}.$$

With a co-prime array using $2M + N$ sensors, the continuous sparse recovery method can detect up to $\frac{MNd}{\lambda}$ sources when $\Delta(\boldsymbol{\theta}) \geq \frac{2\lambda}{MNd}$. The minimum distance constraint is a sufficient condition. In real applications we can expect a more relaxed distance condition for the sources. We will confirm this point in the numerical results. With the utilization of co-prime arrays, the same number of sensors can detect $O(MN)$ sources as indicated by traditional MUSIC theory [48]. We will show in the numerical examples that implementing the super resolution framework provides more degrees of freedom and finer resolution ability than those of MUSIC. This is because the spatial smoothing in MUSIC reduces the obtained virtual array aperture. For the noiseless case, other methods, such as Prony's method [60] and matrix pencil [61] can be used for exact recovery of $O(MN)$ sources. However, they require prior information about the system order, which we do not require here. Furthermore, these methods are generally sensitive to noise in the model and therefore do not offer robustness guarantees.

### 3.3.3 Noisy model for continuous sparse recovery

In practice, the covariance matrix $\boldsymbol{R}_{xx}$ in (3.2) is typically unknown, and cannot be estimated exactly unless the number of samples $T$ goes to infinity. Typically the covariance matrix is approximated by the sample covariance:

$$\hat{\boldsymbol{R}}_{xx} = \frac{1}{T}\sum_{t=1}^{T} \boldsymbol{x}(t)\boldsymbol{x}^*(t). \tag{3.14}$$

Subtracting the noise covariance matrix from both sides, we obtain

$$\hat{\boldsymbol{R}}_{xx} - \sigma^2 \boldsymbol{I} = \boldsymbol{A}\boldsymbol{R}_{ss}\boldsymbol{A}^* + \boldsymbol{E}. \tag{3.15}$$

Here $\boldsymbol{R}_{ss}$ is a diagonal matrix with $k$-th diagonal element

$$\hat{\sigma}_k^2 = \frac{1}{T}\sum_{t=1}^{T} s_k(t)s_k^*(t), \tag{3.16}$$

and the $(m,n)$th element of $\boldsymbol{E}$ is given by (see (3.1))

$$\begin{aligned} E_{mn} = &\frac{1}{T}\sum_{t=1}^{T}\sum_{i,j=1,i\neq j}^{K} A_{mi}A_{nj}^* s_i(t)s_j^*(t) \\ &+ \frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{K} A_{mi}s_i(t)\varepsilon_n^*(t) + \frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{K} \varepsilon_m(t)s_i^*(t)A_{ni}^* \\ &+ \frac{1}{T}\sum_{t=1}^{T} \varepsilon_m(t)\varepsilon_n^*(t) - \sigma^2\boldsymbol{I}_{mn}, \quad 1 \le m,n \le L. \end{aligned} \tag{3.17}$$

For simplicity of analysis, we assume that $\varepsilon_j \sim \mathcal{CN}(0,\sigma^2)$ and $s_i(t) \sim \mathcal{CN}(0,\sigma_s^2)$.

Similar to the operation in (3.3), vectorizing (3.15) leads to

$$\boldsymbol{z} = \mathrm{vec}(\hat{\boldsymbol{R}}_{\boldsymbol{xx}}) = \boldsymbol{\Phi}(\theta_1,\theta_2,\ldots,\theta_K)\boldsymbol{s} + \sigma^2\boldsymbol{1}_n + \boldsymbol{\eta}, \tag{3.18}$$

where $\boldsymbol{\eta} = \text{vec}(\boldsymbol{E})$ and $\boldsymbol{s} = [\hat{\sigma}_1^2, \ldots, \hat{\sigma}_K^2]^{\text{T}}$. For co-prime arrays, by removing repeated rows in (3.18), and sorting them as consecutive lags from $-MNd$ to $MNd$, we get

$$\tilde{\boldsymbol{z}} = \tilde{\boldsymbol{\Phi}}\boldsymbol{s} + \sigma^2\tilde{\boldsymbol{w}} + \tilde{\boldsymbol{e}}, \tag{3.19}$$

in which $\tilde{\boldsymbol{\Phi}}$, and $\tilde{\boldsymbol{w}}$ are defined in (3.5). The vector $\tilde{\boldsymbol{e}}$ is obtained after rearranging $\boldsymbol{\eta}$, and only one element from $\tilde{\boldsymbol{e}}$ corresponds to the diagonal element from $\boldsymbol{E}$. Applying the transformation technique in (3.12), we have

$$\boldsymbol{r} = \boldsymbol{F}s + \boldsymbol{e}, \tag{3.20}$$

where $e(n) = \tilde{e}(n)e^{-\boldsymbol{\jmath}2\pi n\frac{d}{\lambda}}$, and $s$ is the measure defined in (3.7) with $s_k = \hat{\sigma}_k^2$. Thus we can formulate the following continuous sparse recovery problem, which considers the noise:

$$\min_{s} \|s\|_{\text{TV}} \quad \text{s.t.} \quad \|\boldsymbol{F}s - \boldsymbol{r}\|_2 \leq \epsilon. \tag{3.21}$$

This optimization can be solved by first solving the dual problem [53]:

$$\begin{aligned} \max_{\boldsymbol{u},\boldsymbol{Q}} \quad & \text{Re}[\boldsymbol{u}^*\boldsymbol{r}] - \epsilon\|\boldsymbol{u}\|_2 \\ \text{s.t.} \quad & \begin{bmatrix} \boldsymbol{Q} & \boldsymbol{u} \\ \boldsymbol{u}^* & 1 \end{bmatrix} \succeq 0, \\ & \sum_{i=1}^{2MN+1-j} \boldsymbol{Q}_{i,i+j} = \begin{cases} 1 & j = 0, \\ 0 & j = 1, 2, \ldots, 2MN. \end{cases} \end{aligned} \tag{3.22}$$

As before, the primal solution is obtained through a combined process of root finding and least-squares [53].

In order to analyze the robustness of the proposed approach for co-prime arrays, we introduce a lemma that shows that the probability of every element in $\boldsymbol{e}$ being larger than a constant is upper bounded. The proof can be found in the Appendix B.

**Lemma 3.1.** *Let $E_{mn}$ be given in* (3.17) *and assume that $\varepsilon_j \sim \mathcal{CN}(0, \sigma^2)$ and $s_i(t) \sim \mathcal{CN}(0, \sigma_s^2)$ . Then for $m \neq n$, we have*

$$\Pr(|E_{mn}| \geq \epsilon) \leq 8\exp(-C_1(\epsilon)T) + 16\exp(-C_2(\epsilon)T) + 8\exp(-C_3(\epsilon)T).$$

*When $m = n$, and $0 \leq \epsilon \leq 16\sigma^2$, we obtain*

$$\Pr(|E_{mn}| \geq \epsilon) \leq 8\exp(-C_1(\epsilon)T) + 16\exp(-C_2(\epsilon)T) + 4\exp(-C_4(\epsilon)T).$$

*Here $C_1(\epsilon), C_2(\epsilon), C_3(\epsilon)$ and $C_4(\epsilon)$ are increasing functions of $\epsilon$.*

The work of [54] provides an error bound on the support set estimation using (3.21) with noisy measurements. Combining the result from [54] with Lemma 3.1, we have the following theorem.

**Theorem 3.2.** *Consider a co-prime array consisting of two linear arrays with $N$ and $2M$ sensors respectively. The distances between two consecutive sensors are $Md$ for the first array and $Nd$ for the second array, where $M$ and $N$ are co-prime numbers, and $d \leq \frac{\lambda}{2}$. Assume $T$ sample points are collected for each receiver. Suppose we have $K$ sources located at $\theta_1, \ldots, \theta_K$. The minimum distance $\Delta(\boldsymbol{\theta}) \geq \frac{2\lambda}{MNd}$. Let $s(\tau) = \sum_{k=1}^{K} s_k \delta_{\tau_k}$ with $\tau_k = \frac{d}{\lambda}(1 - \sin(\theta_k))$ and $s_k = \hat{\sigma}_k^2$. Consider applying the transformation in* (3.12) *and solving the optimization* (3.21) *with $\epsilon \leq 16\sqrt{2MN+1}\sigma^2$, and denote $s_{\text{opt}}$ as the optimal solution, so that*

$$s_{\text{opt}} = \sum_{\tau_{\text{est}}[i] \in \mathcal{T}_{\text{est}}} s_{\text{est}}[i]\delta_{\tau_{\text{est}}[i]}. \tag{3.23}$$

*Then, for every $\tau_k \in \mathcal{T}$*

$$\left| s_k - \sum_{|\tau_{\text{est}}[i] - \tau_k| \leq \frac{c}{MN}} s_{\text{est}}[i] \right| \leq C_1\epsilon, \tag{3.24}$$

$$\sum_{|\tau_{\text{est}}[i] - \tau_k| \leq \frac{c}{MN}} |s_{\text{est}}[i]|(\tau_{\text{est}}[i] - \tau_k)^2 \leq C_2 \frac{\epsilon}{M^2N^2}, \tag{3.25}$$

*and*

$$\sum_{\tau_k \in \mathcal{T}} \sum_{|\tau_{\text{est}}[i] - \tau_k| > \frac{c}{MN}} |s_{\text{est}}[i]| \leq C_3\epsilon, \tag{3.26}$$

*with probability at least $1 - \alpha e^{-\gamma(\epsilon)T}$, where $\gamma(\epsilon)$ is an increasing function of $\epsilon$. Here $C_1, C_2$ and $C_3$ are positive constants, $c = 0.1649$, and $\mathcal{T} = \{\tau_k : 1 \leq k \leq K\}$ is the support set of the original measure $s$.*

**Proof:** Since $d \leq \frac{\lambda}{2}$, we have that $\tau_k \in [0,1]$ for all $k$ after transformation (3.12). It was shown in [54] that in order to obtain (3.24)-(3.26) we only need to show that $\|\boldsymbol{e}\|_2 \leq \epsilon$ with a certain probability in (3.21). Thus the statistical behavior of $\boldsymbol{e}$ in (3.20) is analyzed first.

Note that

$$\begin{aligned}
&\Pr(\|\boldsymbol{e}\|_2 \leq \epsilon) \geq \Pr\left(\cap_{n=-MN}^{MN} |e(n)| \leq \frac{\epsilon}{\sqrt{2MN+1}}\right) \\
&= 1 - \Pr\left(\cup_{n=-MN}^{MN} |e(n)| \geq \frac{\epsilon}{\sqrt{2MN+1}}\right) \\
&\geq 1 - \sum_{n=-MN}^{MN} \Pr\left(|e(n)| \geq \frac{\epsilon}{\sqrt{2MN+1}}\right), \qquad (3.27)
\end{aligned}$$

which leads to the inequality

$$\begin{aligned}
\Pr(\|\boldsymbol{e}\|_2 \geq \epsilon) &\leq \sum_{n=-MN}^{MN} \Pr\left(|e(n)| \geq \frac{\epsilon}{\sqrt{2MN+1}}\right) \\
&= \sum_{n=-MN}^{MN} \Pr\left(|\tilde{e}(n)| \geq \frac{\epsilon}{\sqrt{2MN+1}}\right). \qquad (3.28)
\end{aligned}$$

The equality follows from the fact that $|e(n)| = |\tilde{e}(n)|$ according to (3.19) and (3.20). Recall that $2MN$ elements of $\tilde{\boldsymbol{e}}$ are taken from $E_{mn}$ when $m \neq n$, and one element of $\tilde{\boldsymbol{e}}$ is taken from $E_{mn}$ when $m = n$. Therefore, by applying the results of Lemma 3.1, we can show that $\|\boldsymbol{F}\boldsymbol{s} - \boldsymbol{r}\|_2 = \|\boldsymbol{e}\|_2 \leq \epsilon$ with probability at least $1 - \alpha e^{-\gamma(\epsilon)T}$, and $\gamma(\epsilon)$ is a increasing function of $\epsilon$. $\square$

Equations (3.24) and (3.25) show that the estimated support set clusters tightly around the true support, while (3.26) indicates that the false peaks in the estimated set $\mathcal{T}_{\text{est}}$ have small amplitudes. A numerical method is proposed in the next section to further refine the estimation, using a discrete sparse recovery method after obtaining $\mathcal{T}_{\text{est}}$.

When both DOAs and signal powers are of interest, we combine the statistical analysis of the noise structure in co-prime arrays with the super resolution results in [53] to give a performance guarantee on the reconstruction of the sparse measure $s$. Since $s$ is a sparse measure, there is no point in bounding $s - s_{\text{opt}}$ directly. Instead $K_h$, which is a low pass filter with cut-off frequency $f_h > MN$, is introduced. This kernel is referred to as the Fejér

kernel, and is given by

$$K_h(t) = \frac{1}{f_h} \sum_{k=-f_h}^{f_h} (f_h + 1 - |k|) e^{\jmath 2\pi k t} = \frac{1}{f_h + 1} \left( \frac{\sin(\pi(f_h + 1)t)}{\sin(\pi t)} \right). \tag{3.29}$$

The cut-off frequency $f_h$ can be much higher than $MN$. Thus using $K_h(t)$, we can show that by solving the convex optimization problem in (3.21) the high resolution details of the original measure $s(\tau) = \sum_{k=1}^{K} s_k \delta_{\tau_k}$ can be recovered with high probability, even though the sample size $T$ is finite.

**Theorem 3.3.** *Let the co-prime arrays and the locations of the sources have the same setup as in Theorem 3.2. The solution of the convex optimization* (3.21) *satisfies*

$$\|K_h * (s_{\text{opt}} - s)\|_{L_1} \leq C_0 \frac{f_h^2}{M^2 N^2} \epsilon, \tag{3.30}$$

*with probability at least* $1 - \alpha e^{-\gamma(\epsilon) T}$ *when* $\epsilon \leq 16\sqrt{2MN + 1}\sigma^2$*, where* $\gamma(\epsilon)$ *is a increasing function of* $\epsilon$*. Here* $C_0$ *is a positive constant number.*

The proof can be obtained by combining Lemma 3.1 with the techniques in [53]. Theorem 3.3 allows to choose the cut-off frequency $f_h$ as large as one wants in order to bound the reconstruction error up to a certain resolution. However, this will entail an increase in the reconstruction error which is proportional to $f_h^2$. This theorem also shows that the reconstruction of $s$ is stable in the presence of noise. The probability of successful reconstruction goes to one exponentially fast as the number of samples $T$ goes to $\infty$. When fixing the probability of a stable reconstruction, by increasing the number of samples $T$ we can allow for a decreased $\epsilon$ since $\gamma(\epsilon)$ is an increasing function. Therefore we can have a larger $f_h$ without increasing the error bound in (3.30). By collecting more samples, one can stably reconstruct the measure $s$ as if we had an even wider aperture $f_h$.

## 3.4 Numerical algorithms

We now derive an optimization framework to reconstruct $s$ for co-prime arrays. Since the optimization is performed on a continuous domain, we will refer to the proposed algorithm as the continuous sparse recovery in the rest of this chapter.

### 3.4.1 DOA estimation with semidefinite programming and root finding

For DOA estimation the noise power $\sigma^2$ is often unknown. Therefore, the optimization must be modified to include this effect. A more realistic optimization is reformulated as

$$\min_{s,\sigma^2\geq 0} \|s\|_{\mathrm{TV}} \quad \text{s.t.} \quad \|\boldsymbol{r} - \boldsymbol{F}s - \sigma^2\boldsymbol{w}\|_2 \leq \epsilon, \tag{3.31}$$

in which $w_n = \tilde{w}_n e^{-\boldsymbol{j}2\pi n\frac{d}{\lambda}}$, and $\tilde{\boldsymbol{w}}$ is defined in (3.5). The dual problem takes on the form

$$\begin{aligned} &\max_{\boldsymbol{u}\in\mathbb{C}^{2MN+1}} \quad \mathrm{Re}[\boldsymbol{u}^*\boldsymbol{r}] - \epsilon\|\boldsymbol{u}\|_2 \\ &\text{s.t.} \quad \|\boldsymbol{F}^*\boldsymbol{u}\|_{L_\infty} \leq 1, \mathrm{Re}[\boldsymbol{u}^*\boldsymbol{w}] \leq 0. \end{aligned} \tag{3.32}$$

The derivation of (3.32) is given in the Appendix C. Since $\boldsymbol{u} = \boldsymbol{0}$ is a feasible solution, strong duality holds according to the general Slater condition [32].

Due to the first constraint in (3.32), the problem itself is still an infinite dimensional optimization. It was shown in [22] that the first constraint can be recast as a semidefinite matrix constraint. Thus the infinite dimensional dual problem is equivalent to the following

semidefinite program (SDP):

$$\begin{aligned} \max_{\boldsymbol{u},\boldsymbol{Q}} \quad & \mathrm{Re}[\boldsymbol{u}^*\boldsymbol{r}] - \epsilon\|\boldsymbol{u}\|_2 \\ \text{s.t.} \quad & \begin{bmatrix} \boldsymbol{Q} & \boldsymbol{u} \\ \boldsymbol{u}^* & 1 \end{bmatrix} \succeq 0, \quad \mathrm{Re}[\boldsymbol{u}^*\boldsymbol{w}] \leq 0, \\ & \sum_{i=1}^{2MN+1-j} \boldsymbol{Q}_{i,i+j} = \begin{cases} 1 & j = 0, \\ 0 & j = 1, 2, \ldots, 2MN. \end{cases} \end{aligned} \tag{3.33}$$

Here $\boldsymbol{Q} \in \mathbb{C}^{(2MN+1)\times(2MN+1)}$ is a Hermitian matrix. This optimization problem can be easily solved, for example by using the CVX package [32], to yield the optimal dual solution.

The following lemma is introduced to link the solutions of the primal and dual problems.

**Lemma 3.2.** *Let $s_{\mathrm{opt}}$ and $\boldsymbol{u}_{\mathrm{opt}} \in \mathbb{C}^{2MN+1}$ be the optimal solutions of the primal problem* (3.31) *and dual problem* (3.33) *respectively. Then*

$$(\boldsymbol{F}^*\boldsymbol{u}_{\mathrm{opt}})(\tau) = \mathrm{sgn}(s_{\mathrm{opt}}(\tau)) \tag{3.34}$$

*for all $\tau$ such that $s_{\mathrm{opt}}(\tau) \neq 0$. Here $\boldsymbol{F}^*$ is the adjoint operator of $\boldsymbol{F}$, and it transforms a vector into a continuous signal by taking the inverse Fourier transform.*

**Proof:** Let $\sigma^2_{\mathrm{opt}}$ be the noise power estimated in the primal problem. Since strong duality holds, we have

$$\begin{aligned} \|s_{\mathrm{opt}}\|_{\mathrm{TV}} &= \mathrm{Re}\langle \boldsymbol{r}, \boldsymbol{u}_{\mathrm{opt}}\rangle - \epsilon\|\boldsymbol{u}_{\mathrm{opt}}\|_2 \\ &= \mathrm{Re}\langle \boldsymbol{r} - \boldsymbol{F}s_{\mathrm{opt}} - \sigma^2_{\mathrm{opt}}\boldsymbol{w}, \boldsymbol{u}_{\mathrm{opt}}\rangle - \epsilon\|\boldsymbol{u}_{\mathrm{opt}}\|_2 + \mathrm{Re}\langle \boldsymbol{F}s_{\mathrm{opt}} + \sigma^2_{\mathrm{opt}}\boldsymbol{w}, \boldsymbol{u}_{\mathrm{opt}}\rangle \\ &\leq \mathrm{Re}\langle \boldsymbol{F}s_{\mathrm{opt}} + \sigma^2_{\mathrm{opt}}\boldsymbol{w}, \boldsymbol{u}_{\mathrm{opt}}\rangle \leq \mathrm{Re}\langle \boldsymbol{F}s_{\mathrm{opt}}, \boldsymbol{u}_{\mathrm{opt}}\rangle. \end{aligned} \tag{3.35}$$

The first inequality follows from the Cauchy-Schwarz inequality and the fact that $\|\boldsymbol{r} - \boldsymbol{F}s_{\mathrm{opt}} - \sigma^2_{\mathrm{opt}}\boldsymbol{w}\|_2 \leq \epsilon$. The second inequality results from $\mathrm{Re}[\boldsymbol{u}^*_{\mathrm{opt}}\boldsymbol{w}] \leq 0$. In addition, we also have

$$\mathrm{Re}\langle \boldsymbol{F}s_{\mathrm{opt}}, \boldsymbol{u}_{\mathrm{opt}}\rangle \leq \|\boldsymbol{F}^*\boldsymbol{u}_{\mathrm{opt}}\|_{L_\infty}\|s_{\mathrm{opt}}\|_{\mathrm{TV}} \leq \|s_{\mathrm{opt}}\|_{\mathrm{TV}}, \tag{3.36}$$

where we used the fact that $\|\boldsymbol{F}^*\boldsymbol{u}_{\mathrm{opt}}\|_{L_\infty} \leq 1$. Combining (3.35) and (3.36) leads to $\|s_{\mathrm{opt}}\|_{\mathrm{TV}} = \mathrm{Re}\langle s_{\mathrm{opt}}, \boldsymbol{F}^*\boldsymbol{u}_{\mathrm{opt}}\rangle$, which implies (3.34). □

According to Lemma 3.2, the supports of $s_{\text{opt}}(\tau)$ satisfy (3.34), and thus can be retrieved by root-finding based on the trigonometric polynomial $1-|\boldsymbol{F}^*\boldsymbol{u}_{\text{opt}}(\tau)|^2=0$. Let $\mathcal{T}_{\text{est}}$ denote the recovered set of roots of this polynomial with cardinality $K_{\text{est}}$, and let $\tau_{\text{est}}[i]$ denote elements in $\mathcal{T}_{\text{est}}$ with $1\leq i\leq K_{\text{est}}$. A matrix $\boldsymbol{F}_{\text{est}}\in\mathbb{C}^{(2MN+1)\times K_{\text{set}}}$ can then be formulated, with measurement $\boldsymbol{r}$ expressed as

$$\boldsymbol{r}=\boldsymbol{F}_{\text{est}}\boldsymbol{s}_0+\sigma^2\boldsymbol{w}+\boldsymbol{e}, \tag{3.37}$$

in which $\boldsymbol{s}_0\in\mathbb{R}^{K_{\text{est}}}$ and

$$\boldsymbol{F}_{\text{est}}=\begin{bmatrix} e^{-\boldsymbol{j}MNd2\pi\tau_{\text{est}}[1]} & \dots & e^{-\boldsymbol{j}MNd2\pi\tau_{\text{est}}[K_{\text{est}}]} \\ e^{-\boldsymbol{j}(MN-1)d2\pi\tau_{\text{est}}[1]} & \dots & e^{-\boldsymbol{j}(MN-1)d2\pi\tau_{\text{est}}[K_{\text{est}}]} \\ \vdots & \ddots & \vdots \\ e^{\boldsymbol{j}MNd2\pi\tau_{\text{est}}[1]} & \dots & e^{\boldsymbol{j}MNd2\pi\tau_{\text{est}}[K_{\text{est}}]} \end{bmatrix}.$$

Due to numerical issues in the root finding process, the cardinality of $\mathcal{T}_{\text{est}}$ is normally larger than the cardinality of $\mathcal{T}$, i.e., $K_{\text{est}}\geq K$. It is possible in some cases that $K_{\text{est}}\geq 2MN+1$, which would lead to an ill-conditioned linear system (3.37). Sparsity can then be exploited on $\boldsymbol{s}_0$. A convex optimization in the discrete domain can be formulated as

$$\min_{\boldsymbol{s}_0,\sigma^2\geq 0}\|\boldsymbol{s}_0\|_1 \quad \text{s.t.} \quad \|\boldsymbol{r}-\boldsymbol{F}_{\text{est}}\boldsymbol{s}_0-\sigma^2\boldsymbol{w}\|_2\leq\epsilon_d. \tag{3.38}$$

We choose $\epsilon_d$ in (3.38) to be larger than $\epsilon$ in (3.31) since the noise level is expected to be higher in (3.37) due to inevitable error introduced in the root finding process. Assuming that the solution of (3.38) is $\boldsymbol{s}_{\text{est}}\in\mathbb{R}^{K_{\text{est}}}$, the estimation of the measure $s(\tau)$ in the continuous domain can be represented as

$$s_{\text{opt}}(\tau)=\sum_{i=1}^{K_{\text{est}}}s_{\text{est}}[i]\delta_{\tau_{\text{est}}[i]}. \tag{3.39}$$

### 3.4.2 Extension: source number detection

Conventional source number detection for array processing is typically performed by exploiting eigenvalues from the sample covariance matrix. For co-prime arrays, this covariance matrix can be obtained by performing spatial smoothing on $\tilde{\boldsymbol{z}}$ in (3.5). The same idea can

also be implemented on the sparse signal $\boldsymbol{s}_{\text{est}}$ recovered from the previous section. Ideally, after sorting its elements in a descending order, the signal $\boldsymbol{s}_{\text{est}}$ reconstructed from (3.38) should follow

$$s^2_{\text{est}}[1] \geq s^2_{\text{est}}[2] \geq \ldots s^2_{\text{est}}[K] \geq s^2_{\text{est}}[K+1] = \ldots = s^2_{\text{est}}[K_{\text{est}}] = 0. \tag{3.40}$$

The SORTE algorithm can be applied to this series. The difference of the elements from $\boldsymbol{s}_{\text{est}}$ is

$$\nabla s_{\text{est}}[i] = s^2_{\text{est}}[i] - s^2_{\text{est}}[i+1],\ i = 1, \ldots, K_{\text{est}} - 1. \tag{3.41}$$

The gap measure in SORTE is given as

$$\text{SORTE}(i) = \begin{cases} \frac{\text{var}[i+1]}{\text{var}[i]} & \text{var}[i] \neq 0, \\ +\infty & \text{var}[i] = 0, \end{cases} \quad i = 1, \ldots, K_{\text{est}} - 2, \tag{3.42}$$

where

$$\text{var}[i] = \frac{1}{K_{\text{est}} - i} \sum_{m=i}^{K_{\text{est}}-1} \left( \nabla s_{\text{est}}[m] - \frac{1}{K_{\text{est}} - i} \sum_{n=i}^{K_{\text{est}}-1} \nabla s_{\text{est}}[n] \right)^2. \tag{3.43}$$

The number of sources can be estimated as

$$\hat{K} = \text{argmin}_i \quad \text{SORTE}(i). \tag{3.44}$$

This approach requires $K_{\text{est}} > 2$ due to the definition of SORTE$(i)$ in (3.42). When $K_{\text{est}} \leq 2$, since $\mathcal{T}_{\text{est}}$ is obtained from the rooting finding process based on the continuous sparse recovery, we simply let $\hat{K} = K_{\text{est}}$. We will refer to this continuous sparse recovery based SORTE as CSORTE.

## 3.5 Numerical examples

In this section, we present several numerical examples to show the merits of implementing our continuous sparse recovery techniques to co-prime arrays.

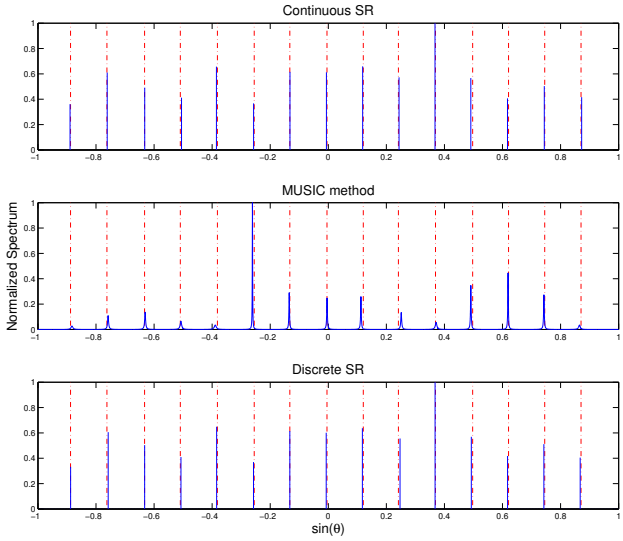


Figure 3.2: Normalized spectra for CSR, MUSIC, and DSR, with $T = 500$ and SNR$=-10$dB.

We consider a co-prime array with 10 sensors. One set of sensors is located at positions $[0, 3, 6, 9, 12]d$, and the second set is located at $[0, 5, 10, 15, 20, 25]d$, where $d$ is taken as half of the wavelength. The first sensors from both sets are collocated. It is easy to show that the correlation matrix generates a virtual array with lags from $-17d$ to $17d$. We compare continuous sparse recovery (CSR) techniques with MUSIC and also with discrete sparse recovery method (DSR) considering grid mismatches [52]. In [52], a LASSO formulation is used to perform the DOA estimation. Here we implement an equivalent form of LASSO, i.e., Basis Pursuit, to perform the comparison. The MUSIC method in this simulation follows the spatial smoothing technique in [48]. For the discrete sparse recovery method, we take the grid from $-1$ to 1, with step size 0.005 for $\sin(\theta)$. The noise levels $\epsilon$ in the optimization formulas are chosen by cross validation. We consider 15 narrow band signals located at $\sin(\boldsymbol{\theta}) = [-0.8876, -0.7624, -0.6326, -0.5096, -0.3818, -0.2552, -0.1324, -0.0046, 0.1206, 0.2414, 0.3692, 0.4972, 0.6208, 0.7454, 0.8704]$. We show that continuous sparse recovery yields better results in terms of detection ability, resolution, and estimation accuracy.

### 3.5.1 Degrees of freedom

In this first numerical example, we verify that the proposed continuous sparse recovery increases the degrees of freedom to $O(MN)$ by implementing the co-prime arrays' structure. The number of time samples is 500 and the SNR is chosen to be $-10$dB. The $\epsilon$ for CSR is taken as 5, and $\epsilon_d$ is taken as 10 while DSR uses $\epsilon = 10$. In Fig. 3.2, we use a dashed line to represent the true directions of arrival. The CPU time for running CSR was 7.30 seconds. DSR took 7.82 seconds, whereas MUSIC algorithm took only 0.81 seconds. In MUSIC, we implement a root MUSIC algorithm to estimate the location of each source, where the number of sources is assumed to be given. The average estimation errors for CSR, DSR, and root MUSIC are $0.23\%, 0.26\%$, and $0.42\%$ respectively. We can see that all three methods achieve $O(MN)$. In the following subsection, we test the estimation accuracy of these three methods via Monte Carlo simulations.

### 3.5.2 Estimation accuracy

In this section, we compare CSR, DSR and MUSIC via Monte Carlo simulations. Since traditional MUSIC does not yield the DOA of each source directly, we consider the Root MUSIC algorithm instead. For simplicity, we will still refer to it as MUSIC in this section. The number of sources is assumed to be known for the MUSIC algorithm in this simulation, whereas sparse methods do not assume this a priori. The values of $\epsilon$ and $\epsilon_d$ are chosen to be 5 and 10, while discrete SR uses $\epsilon = 10$.

Figure 3.3 shows the DOA estimation error as a function of SNR after 50 Monte Carlo simulations. The estimation error is calculated based on the sine function of the DOAs. The average CPU times for running CSR, DSR and MUSIC are 6.93s, 9.30s, and 1.46s respectively. We can see that CSR performs better than DSR uniformly with less computing time. Both sparse recovery methods achieve better DOA estimation accuracy than MUSIC. The accuracy of DSR can be further improved by taking a finer grid with a smaller step-size. However, this will slow down DSR further.

In Fig. 3.4 we show that with a varying number of snapshots the proposed CSR also exhibits better estimation accuracy than either DSR or MUSIC. The average CPU times for running

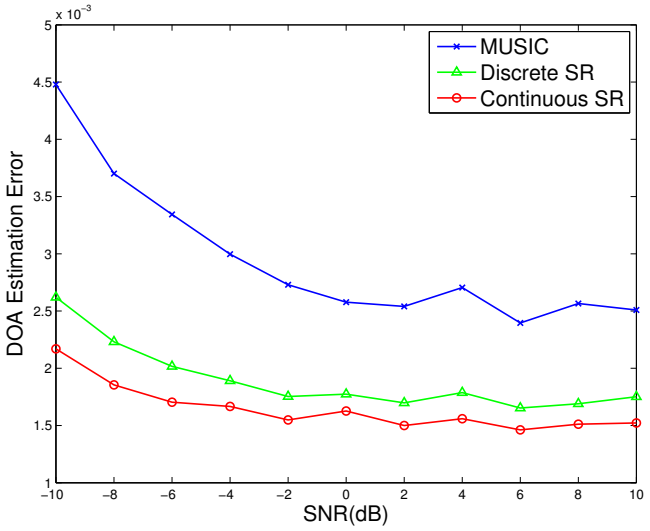


Figure 3.3: DOA estimation errors for CSR, MUSIC, and DSR, with $T = 500$.

CSR, DSR and MUSIC are 6.50s, 7.91s, and 1.43s respectively. The performance of MUSIC and DSR approaches the performance of CSR when the number of snapshots is close to 5000. We can see that implementing CSR can save sampling time by taking a small number of snapshots to achieve the same estimation accuracy as the MUSIC algorithm. The parameters $\epsilon$ and $\epsilon_d$ are equal to 5 and 10 in this simulation.

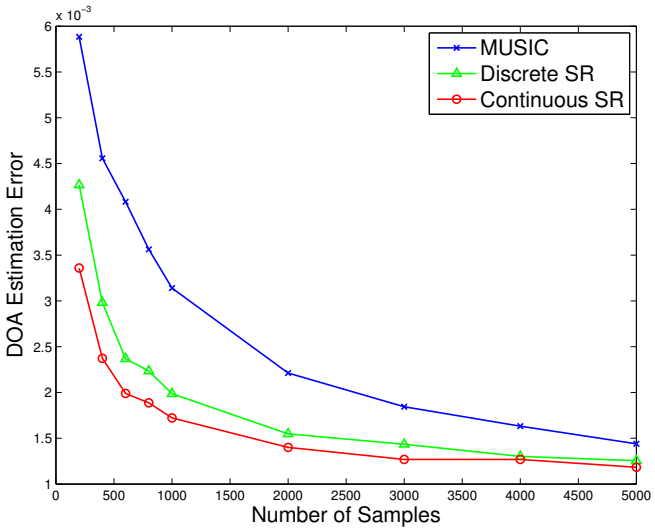


Figure 3.4: DOA estimation error for CSR, MUSIC, and DSR, with SNR=$-10$ dB.

### 3.5.3 Source number detection performance comparsion

We now compare the source number detection performance of the proposed CSORTE with that of traditional SORTE applied to the covariance matrix after spatial smoothing. The SNR is set to 0dB while the number of snapshots is 3000. We vary the number of sources from 11 to 17. Since this co-prime array structure yields consecutive lags from $-17d$ to $17d$, 17 is the maximum number of sources that can be detected theoretically via techniques based on the covariance matrix.

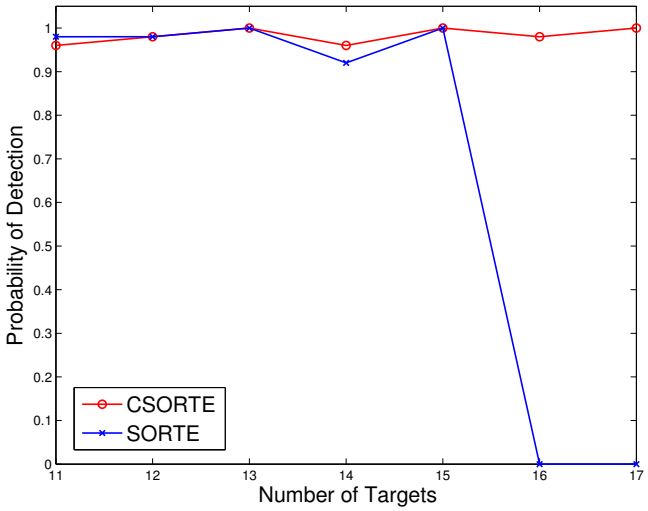


Figure 3.5: Source number detection using CSORTE and SORTE, with SNR=0 dB, $T = 3000$.

Figure 3.7 shows the probability of detection with respect to the number of sources after 50 Monte Carlo simulations. In CSR, $\epsilon$ is chosen to be $5\sigma$, and $\epsilon_d$ is set to be $2\epsilon$. When the number of sources is less than 15, CSORTE and SORTE yield comparable result. However, SORTE fails after the number of sources is larger than 15, while CSORTE provides stable performance and also exhibits perfect detection even when the number of sources reaches the theoretical limit of 17. DSR can also be combined with SORTE to perform source number detection. However, the detection accuracy is jeopardized by the spurious signal from the reconstructed signals using DSR. Therefore SORTE based on DSR is not included here.

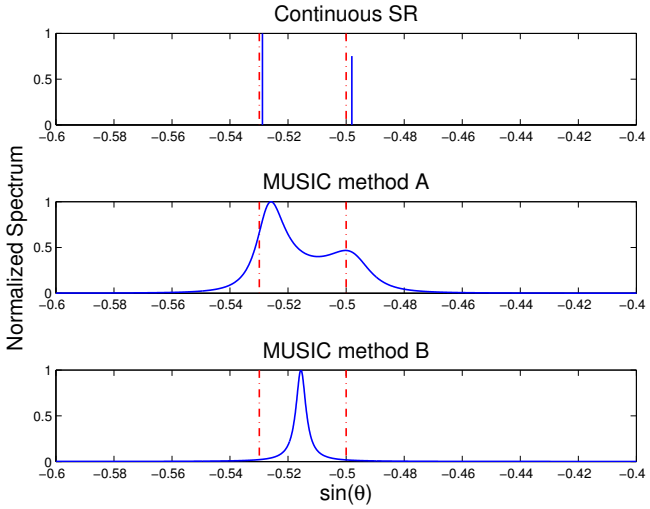


Figure 3.6: Source number detection using CSR and the MUSIC algorithm, with SNR=0 dB, $T = 500$.

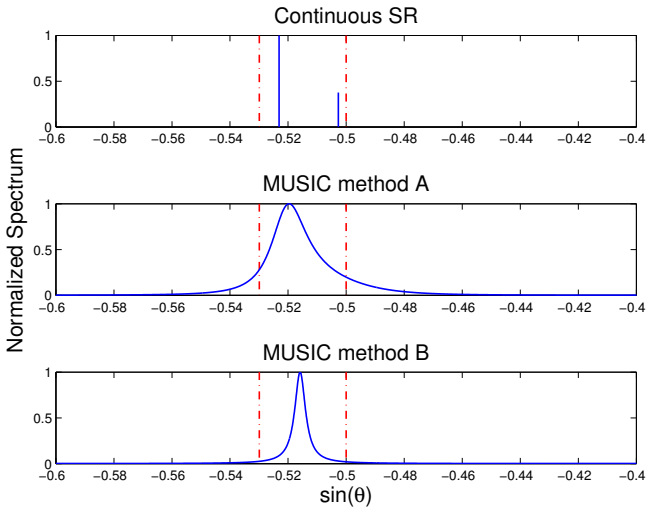


Figure 3.7: Source number detection using CSR and MUSIC algorithm, with SNR=$-5$ dB, $T = 500$.

### 3.5.4 Resolution ability

Finally we compare the resolution abilities of CSR and MUSIC, and show that CSR is capable of resolving very closely located signals. In the first simulation, two sources are

closely located at $-32°$ and $-30°$.The value of $\epsilon$ is chosen to be $0.7\sigma$ and $\epsilon_d$ is set to be $2\epsilon$ in CSR, where $\sigma$ is the noise power.

Figure 3.6 shows a numerical example in which the SNR is 0 dB and the number of snapshots is 500. Normalized spectra are plotted for three methods. MUSIC method A is the MUSIC algorithm with the assumption that the number of sources is known while the MUSIC method B is MUSIC relying on traditional SORTE to provide the estimated number of sources. We can see that MUSIC method B fails to resolve these two targets because traditional SORTE fails to estimate the number of sources correctly. CSR resolves the two sources successfully even though a priori information about the number of sources is not assumed to be given. In Fig. 3.7, we lower the SNR to $-5$ dB, and we notice that even given the number of sources, MUSIC fails to resolve the two closely located sources while CSR resolves them successfully.

Note, that while a separation of $\frac{2\lambda}{MNd}$ is sufficient for Theorem III.1 to hold, in real applications, we expect to observe a better result. Thus we intentionally chose two sources which are more closely located, to show that the proposed method still works even with a stronger constraint.

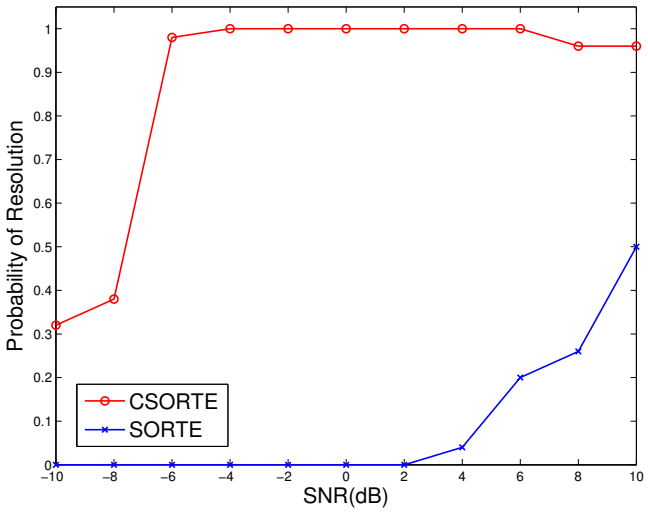


Figure 3.8: Comparison of resolution performance of CSORTE and SORTE, with $T = 2000$.

Finally we conduct a simulation based on Monte Carlo runs to compare the resolution ability of CSORTE and the traditional SORTE algorithm. Figure 3.8 shows the resolution performance in detecting two sources located at $-32°$ and $-30°$, using CSORTE and SORTE

after 50 Monte Carlo runs. The parameter $\epsilon$ is chosen to be $0.7\sigma$, and $\epsilon_d$ is set to be $2\epsilon$ in CSR. We can see that CSORTE outperforms traditional SORTE when detecting the two closely located sources.

## 3.6 Summary

In this chapter we extended sparse recovery to a continuous domain. By doing this, we were able to neglect the effect of dictionary mismatches introduced by off-grid targets. In particular, we extended the recently developed mathematical theory of super resolution to DOA estimation using co-prime arrays. We successfully showed that by using TV-norm minimization and co-prime arrays, we could increase the degrees of freedom from $O(M+N)$ to $O(MN)$. Using a primal-dual approach we transformed the original infinite dimensional TV-norm minimization to a solvable semidefinite program. After estimating the candidate support sets by root finding, we solved a small scale sparse recovery problem. The robustness of the proposed super resolution approach was verified by performing statistical analysis of the noise inherent in co-prime array processing. A source number detection algorithm was then proposed by combining the existing SORTE algorithm with the reconstructed spectrum from continuous sparse recovery. With numerical examples, we showed that the proposed method achieves a better DOA estimation accuracy than the method proposed in Chapter 2, and also exhibits improved resolution ability over traditional MUSIC with spatial smoothing. The same algorithm and analysis can also be performed for nested arrays and collocated MIMO radar.

# Chapter 4

# Smoothing and Decomposition for the Analysis Model of Sparse Recovery

In this chapter we will focus on the analysis model of sparse recovery. Instead of assuming that the original signal can be decomposed into a sparse combination of atoms in a given dictionary, the analysis model assumes that there exists a transformation which will make the original signal sparse. Efficient algorithms and a theoretical performance bound will be derived in this chapter. [3]

## 4.1 Introduction

We consider a typical under-determined recovery problem having the following linear form:

$$\boldsymbol{b} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{w}, \tag{4.1}$$

in which $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ is a measurement matrix, $\boldsymbol{b} \in \mathbb{R}^m$ is the measurement vector, and $\boldsymbol{w} \in \mathbb{R}^m$ represents the noise term. Our goal is to recover the signal $\boldsymbol{x} \in \mathbb{R}^n$, and we have $m < n$, which indicates that the inverse problem is ill-posed and has infinitely many solutions. To find a unique solution, prior information on $\boldsymbol{x}$ must be incorporated.

[3] This chapter is based on Z. Tan,Y. C. Eldar, A. Beck and A. Nehorai,"Smoothing and decomposition for analysis sparse recovery," *IEEE Trans. Signal Processing*, vol. 62, pp. 1762-1774, April 2014. 

In the synthesis approach to sparse recovery from previous chapter, it is assumed that $\boldsymbol{x}$ can be expressed as a sparse combination of known dictionary elements, represented as columns of a matrix $\boldsymbol{D} \in \mathbb{R}^{n\times p}$ with $p \geq n$. That is $\boldsymbol{x} = \boldsymbol{D}\boldsymbol{\alpha}$ with $\boldsymbol{\alpha}$ sparse, i.e., the number of non-zero elements in $\boldsymbol{\alpha}$ is far less than the length of $\boldsymbol{\alpha}$. The main methods for solving this problem can be classified into two categories. One includes greedy methods, such as iterative hard thresholding [10] and orthogonal matching pursuit [11]. The other is based on relaxation-type methods, such as basis pursuit [23] and LASSO [13]. These methods can stably recover a sparse signal $\boldsymbol{\alpha}$ when the matrix $\boldsymbol{AD}$ satisfies the restricted isometry property (RIP) [40]-[63].

Recently, an alternative approach has became popular, which is known as the analysis method [24], [25]. In this framework, we are given an analysis dictionary $\boldsymbol{D}^*(\boldsymbol{D} \in \mathbb{R}^{n\times p})$ under which $\boldsymbol{D}^*\boldsymbol{x}$ is sparse. Assuming, for example, that the $\ell_2$ norm of the noise $\boldsymbol{w}$ is bounded by $\varepsilon$, the recovery problem can be formulated as

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} \|\boldsymbol{D}^*\boldsymbol{x}\|_0 \quad \text{subject to } \|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}\|_2 \leq \varepsilon. \tag{4.2}$$

Since this problem is NP hard, several greedy algorithms have been proposed to approximate it, such as thresholding [27] and subspace pursuit [64].

Alternatively, the nonconvex $\ell_0$ norm can be approximated by the convex $\ell_1$ norm leading to the following relaxed problem, referred to as analysis basis pursuit (ABP):

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} \|\boldsymbol{D}^*\boldsymbol{x}\|_1 \quad \text{subject to } \|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}\|_2 \leq \varepsilon. \tag{4.3}$$

ABP is equivalent to the unconstrained optimization

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} \frac{1}{2}\|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \lambda\|\boldsymbol{D}^*\boldsymbol{x}\|_1, \tag{4.4}$$

which we call analysis LASSO (ALASSO). The equivalence is in the sense that for any $\varepsilon > 0$ there exists a $\lambda$ for which the optimal solutions of ABP and ALASSO are identical.

Both optimization problems ABP and ALASSO can be solved using interior point methods [32]. However, when the problem dimension grows, these techniques become very slow since they require solutions of linear systems. Another suggested approach is based on alternating

direction method of multipliers (ADMM) [65, 66]. The efficiency of this method highly depends on nice structure of the matrices $\boldsymbol{A}$. Fast versions of first-order algorithms, such as the fast iterative shrinkage-thresholding algorithm (FISTA) [14], are more favorable in dealing with large dimensional data since they do not require $\boldsymbol{A}$ to have any structure. The difficulty in directly applying first-order techniques to ABP (4.3) and ALASSO (4.4) is the fact that the nonsmooth term $\|\boldsymbol{D}^*\boldsymbol{x}\|_1$ is inseparable. A generalized iterative soft-thresholding algorithm was proposed in [67] to tackle this difficulty. However, this approach converges relatively slow as we will show in one of our numerical examples. A common alternative is to transform the nondifferentiable problem into a smooth counterpart. In [15], the authors used Nesterov's smoothing-based method [68] in conjunction with continuation (NESTA) to solve ABP (4.3), under the assumption that the matrix $\boldsymbol{A}^*\boldsymbol{A}$ is an orthogonal projector. In [69], a smoothed version of ALASSO (4.4) is solved using a nonlinear conjugate gradient descent algorithm. To avoid imposing conditions on $\boldsymbol{A}$, we focus in this chapter on the ALASSO formulation (4.4).

It was shown in [43] that one can apply any fast first-order method that achieves an $\varepsilon$-optimal solution within $O(\frac{1}{\sqrt{\varepsilon}})$ iterations, to an $\varepsilon$ smooth-approximation of the general nonsmooth problem and obtain an algorithm with $O(\frac{1}{\varepsilon})$ iterations. In this chapter, we choose a monotone version of FISTA (MFISTA) [70] as our fast first-order method, whose objective function values are guaranteed to be non-increasing. We apply the smoothing approach together with MFISTA leading to the smoothing-based MFISTA (SFISTA) algorithm. We also propose a decomposition-based MFISTA method (DFISTA) to solve the analysis sparse recovery problem. The decomposition idea is to introduce an auxiliary variable $\boldsymbol{z}$ in (4.4) so that MFISTA can be applied in a simple and explicit manner. This decomposition approach can be traced back to [71], and has been widely used for solving total variation problems in the context of image reconstruction [72].

Both smoothing and decomposition based algorithms for nonsmooth optimization problems are very popular in the literature. One of the main goals of this chapter is to examine their respective performance. We show that SFISTA requires lower computational complexity to reach a predetermined accuracy. Our results can be applied to a general model, and are not restricted to the analysis sparse recovery problem.

In the context of analysis sparse recovery, we show in Section 4.2.3 that both smoothing and decomposition techniques solve the following optimization problem:

$$\min_{\boldsymbol{x}\in\mathbb{R}^n,\boldsymbol{z}\in\mathbb{R}^p} \frac{1}{2}\|\boldsymbol{A}\boldsymbol{x}-\boldsymbol{b}\|_2^2+\lambda\|\boldsymbol{z}\|_1+\frac{1}{2}\rho\|\boldsymbol{z}-\boldsymbol{D}^*\boldsymbol{x}\|_2^2, \tag{4.5}$$

which we refer to as relaxed ALASSO (RALASSO). Another contribution of this chapter is in proving recovery guarantees for RALASSO (4.5). With the introduction of the restricted isometry property adapted to $\boldsymbol{D}$ (D-RIP) [24], previous work [24] [73] studied recovery guarantees based on ABP (4.3) and ALASSO (4.4). Here we combine the techniques in [40] and [73], and obtain a performance bound on RALASSO (4.5). We show that when $\sigma_{2s}<0.1907$ and $\|\boldsymbol{D}^*\boldsymbol{A}^*\boldsymbol{w}\|_\infty\leq\frac{\lambda}{2}$, the solution $\hat{\boldsymbol{x}}_\rho$ of RALASSO (4.5) satisfies

$$\|\hat{\boldsymbol{x}}_\rho-\boldsymbol{x}\|_2\leq C_0\sqrt{s}\lambda+C_1\frac{\|\boldsymbol{D}^*\boldsymbol{x}-(\boldsymbol{D}^*\boldsymbol{x})_s\|_1}{\sqrt{s}}+C_2\frac{\lambda p}{\sqrt{s}\rho}, \tag{4.6}$$

where $p$ is the number of rows in $\boldsymbol{D}^*$, $C_0, C_1, C_2$ are constants, and we use $(\boldsymbol{x})_s$ to denote the vector consisting of the largest $s$ entries of $|\boldsymbol{x}|$. As a special case, choosing $\rho\to\infty$ extends the bound in (4.6) and obtains the reconstruction bound for ALASSO (4.4) as long as $\sigma_{2s}<0.1907$, which improves upon the results of [73].

The chapter is organized as follows. In Section 4.2, we introduce some mathematical preliminaries, and present SFISTA and DFISTA for solving RALASSO (4.5). We analyze the convergence behavior of these two algorithms in Section 4.3, and show that SFISTA converges faster than DFISTA for a general model. Performance guarantees on RALASSO (4.5) are developed in Section 4.4. Finally, in Section 4.5 we test our techniques on numerical experiments to demonstrate the effectiveness of our algorithms in solving the analysis recovery problem. We show that SFISTA performs favorably in comparison with DFISTA. A continuation method is also introduced to further accelerate the convergence speed.

## 4.2 Smoothing and decomposition relaxation for analysis model

In this section we present the smoothing-based and decomposition-based methods for solving ALASSO (4.4). To do so, we first recall in Subsection 4.2.1 some results related to proximal gradient methods that will be essential to our presentation and analysis.

### 4.2.1 The proximal gradient method

We begin this section with the recall of Moreau's proximal (or "prox") operator and its properties [45], which is the key step in defining the proximal gradient method.

Given a closed proper convex function $h : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$, the proximal operator of $h$ is defined by

$$\text{prox}_h(\boldsymbol{x}) = \underset{\boldsymbol{u}\in\mathbb{R}^n}{\arg\min} \left\{ h(\boldsymbol{u}) + \frac{1}{2}\|\boldsymbol{u} - \boldsymbol{x}\|_2^2 \right\}. \tag{4.7}$$

The proximal operator can be computed efficiently in many important instances. For example, it can be easily obtained when $h$ is an $l_p$ norm ($p \in [1, \infty)$), or an indicator of "simple" closed convex sets such as the box, unit-simplex and the ball. More examples of proximal operators as well as a wealth of properties can be found, for example, in [74] [75].

The proximal operator can be used in order to compute smooth approximations of convex functions. Specifically, let $h$ be a closed, proper, convex function, and let $\mu > 0$ be a given parameter. Define

$$h_\mu(\boldsymbol{x}) = \min_{\boldsymbol{u}\in\mathbb{R}^n} \left\{ h(\boldsymbol{u}) + \frac{1}{2\mu}\|\boldsymbol{u} - \boldsymbol{x}\|_2^2 \right\}. \tag{4.8}$$

It is easy to see that

$$h_\mu(\boldsymbol{x}) = h(\text{prox}_{\mu h}(\boldsymbol{x})) + \frac{1}{2\mu}\|\boldsymbol{x} - \text{prox}_{\mu h}(\boldsymbol{x})\|_2^2. \tag{4.9}$$

The function $h_\mu$ is called the *Moreau envelope of* $h$ and has the following important properties (see [45] for further details):

- $h_\mu(\boldsymbol{x}) \leq h(\boldsymbol{x})$.
- $h_\mu$ is continuously differentiable and its gradient is Lipschitz continuous with constant $1/\mu$.
- The gradient of $h_\mu$ is given by

$$\nabla h_\mu(\boldsymbol{x}) = \frac{1}{\mu}(\boldsymbol{x} - \text{prox}_{\mu h}(\boldsymbol{x})). \tag{4.10}$$

One important usage of the proximal operator is in the proximal gradient method that is aimed at solving the following composite problem:

$$\min_{\boldsymbol{x}\in\mathbb{R}^n}\{F(\boldsymbol{x}) + G(\boldsymbol{x})\}. \tag{4.11}$$

Here $F : \mathbb{R}^n \to \mathbb{R}$ is a continuously differentiable convex function with a continuous gradient that has Lipschitz constant $L_{\nabla F}$:

$$\|\nabla F(\boldsymbol{x}) - \nabla F(\boldsymbol{y})\|_2 \leq L_{\nabla F}\|\boldsymbol{x} - \boldsymbol{y}\|_2, \quad \text{for all } \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n,$$

and $G : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ is an extended-valued, proper, closed and convex function. The *proximal gradient method* for solving (4.11) takes the following form (see [14, 76]):

---

**Proximal Gradient Method For Solving (4.11)**

---

**Input**: An upper bound $L \geq L_{\nabla F}$.
**Step 0.** Take $\boldsymbol{x}_0 \in \mathbb{R}^n$.
**Step k.** $(k \geq 1)$
Compute $\boldsymbol{x}_k = \text{prox}_{\frac{1}{L}G}\left(\boldsymbol{x}_{k-1} - \frac{1}{L}\nabla F(\boldsymbol{x}_{k-1})\right)$.

---

Table 4.1: Proximal gradient method

The main disadvantage of the proximal gradient method is that it suffers from a relatively slow $O(1/k)$ rate of convergence of the function values. An accelerated version is the *fast*

*proximal gradient method*, also known in the literature as *fast iterative shrinkage thresholding algorithm* (FISTA) [14, 76]. When $G \equiv 0$, the problem is smooth, and FISTA coincides with Nesterov's optimal gradient method [77]. In this chapter we implement a monotone version of FISTA (MFISTA) [70], which guarantees that the objective function value is non-increasing along the iterations.

---

**Monotone FISTA Method (MFISTA) For Solving (4.11)**

---

**Input**: An upper bound $L \geq L_{\nabla F}$.
**Step 0.** Take $\boldsymbol{y}_1 = \boldsymbol{x}_0, t_1 = 1$.
**Step k.** ($k \geq 1$) Compute
$\boldsymbol{z}_k = \text{prox}_{\frac{1}{L}G}\left(\boldsymbol{y}_k - \frac{1}{L}\nabla F(\boldsymbol{y}_k)\right)$.
$t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2}$.
$\boldsymbol{x}_k = \text{argmin}\{F(\boldsymbol{x}) + G(\boldsymbol{x}) : \boldsymbol{x} = \boldsymbol{z}_k, \boldsymbol{x}_{k-1}\}$.
$\boldsymbol{y}_{k+1} = \boldsymbol{x}_k + \frac{t_k}{t_{k+1}}(\boldsymbol{z}_k - \boldsymbol{x}_k) + \frac{t_k - 1}{t_{k+1}}(\boldsymbol{x}_k - \boldsymbol{x}_{k-1})$.

---

Table 4.2: Monotone FISTA method (MFISTA)

The rate of convergence of the sequence generated by MFISTA is $O(1/k^2)$.

**Theorem 4.1.** *[70] Let $\{\boldsymbol{x}_k\}_{k\geq 0}$ be the sequence generated by MFISTA, and let $\hat{\boldsymbol{x}}$ be an optimal solution of (4.11). Then*

$$F(\boldsymbol{x}_k) + G(\boldsymbol{x}_k) - F(\hat{\boldsymbol{x}}) - G(\hat{\boldsymbol{x}}) \leq \frac{2L_{\nabla F}\|\boldsymbol{x}_0 - \hat{\boldsymbol{x}}\|_2^2}{(k+1)^2}. \tag{4.12}$$

### 4.2.2 The general nonsmooth model

The general optimization model we consider in this chapter is

$$\min_{\boldsymbol{x}\in\mathbb{R}^n}\{H(\boldsymbol{x}) = f(\boldsymbol{x}) + g(\boldsymbol{D}^*\boldsymbol{x})\}, \tag{4.13}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is a continuously differentiable convex function with a Lipschitz continuous gradient $L_{\nabla f}$. The function $g : \mathbb{R}^p \to \mathbb{R} \cup \{\infty\}$ is a closed, proper convex function which is not necessarily smooth, and $\boldsymbol{D}^* \in \mathbb{R}^{p\times n}$ is a given matrix. In addition, we assume that $g$ is Lipschitz continuous with parameter $L_g$:

$$|g(\boldsymbol{z}) - g(\boldsymbol{v})| \leq L_g \|\boldsymbol{z} - \boldsymbol{v}\|_2 \quad \text{for all } \boldsymbol{z}, \boldsymbol{v} \in \mathbb{R}^p.$$

This is equivalent to saying that the subgradients of $g$ over $\mathbb{R}^p$ are bounded by $L_g$:

$$\|g'(\boldsymbol{z})\|_2 \leq L_g \text{ for any } \boldsymbol{x} \in \mathbb{R}^n \text{ and } g'(\boldsymbol{z}) \in \partial g(\boldsymbol{z}).$$

An additional assumption we make throughout is that the proximal operator of $\alpha g(\boldsymbol{z})$ for any $\alpha > 0$ can be easily computed.

Directly applying MFISTA to (4.13) requires computing the proximal operator of $g(\boldsymbol{D}^*\boldsymbol{x})$. Despite the fact that we assume that it is easy to compute the proximal operator of $g(\boldsymbol{z})$, it is in general difficult to compute that of $\alpha g(\boldsymbol{D}^*\boldsymbol{x})$. Therefore we need to transform the problem before utilizing MFISTA, in order to avoid this computation.

When considering ALASSO, $f(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|_2^2$ and $g(\boldsymbol{D}^*\boldsymbol{x}) = \lambda\|\boldsymbol{D}^*\boldsymbol{x}\|_1$. The Lipschitz constants are given by $L_{\nabla f} = \|\boldsymbol{A}\|_2^2$ and $L_g = \lambda\sqrt{p}$. The proximal operator of $\alpha g(\boldsymbol{z}) = \alpha\lambda\|\boldsymbol{z}\|_1$ can be computed as

$$\text{prox}_{\alpha g}(\boldsymbol{z}) = \Gamma_{\lambda\alpha}(\boldsymbol{z}) = [|\boldsymbol{z}| - \lambda\alpha]_+ \text{sgn}(\boldsymbol{z}), \tag{4.14}$$

where for brevity, we denote the soft shrinkage operator by $\Gamma_{\lambda\alpha}(\boldsymbol{z})$. Here $[\boldsymbol{z}]_+$ denotes the vector whose components are given by the maximum between $z_i$ and 0. Note, however, that there is no explicit expression for the proximal operator of $g(\boldsymbol{D}^*\boldsymbol{x}) = \lambda\|\boldsymbol{D}^*\boldsymbol{x}\|_1$, i.e., there is no closed form solution to

$$\underset{\boldsymbol{u}\in\mathbb{R}^n}{\arg\min} \left\{ \alpha\lambda\|\boldsymbol{D}^*\boldsymbol{u}\|_1 + \frac{1}{2}\|\boldsymbol{u} - \boldsymbol{x}\|_2^2 \right\}. \tag{4.15}$$

In the next subsection, we introduce two popular approaches for transforming the problem (4.13): smoothing and decomposition. We will show in Sections 4.2.4 and 4.2.5 that both

transformations lead to algorithms which only require computation of the proximal operator of $g(\boldsymbol{z})$, and not that of $g(\boldsymbol{D}^*\boldsymbol{x})$.

### 4.2.3 The smoothing and decomposition transformations

The first approach to transform (4.13) is the smoothing method in which the nonsmooth function $g(\boldsymbol{z})$ is replaced by its Moreau envelope $g_\mu(\boldsymbol{z})$, which can be seen as a smooth approximation. By letting $\boldsymbol{z} = \boldsymbol{D}^*\boldsymbol{x}$ , the smoothed problem becomes

$$\min_{\boldsymbol{x}\in\mathbb{R}^n}\{H_\mu(\boldsymbol{x}) = f(\boldsymbol{x}) + g_\mu(\boldsymbol{D}^*\boldsymbol{x})\}, \tag{4.16}$$

to which MFISTA can be applied since it only requires evaluating the proximal operator of $g(\boldsymbol{z})$. From the general properties of the Moreau envelope, and from the fact that the norms of the subgradients of $g$ are bounded above by $L_g$, we can deduce that there exists some $\beta_1$, $\beta_2 > 0$ such that $\beta_1 + \beta_2 = L_g$ and $g(\boldsymbol{z}) - \beta_1\mu \leq g_\mu(\boldsymbol{z}) \leq g(\boldsymbol{z}) + \beta_2\mu$ for all $\boldsymbol{z} \in \mathbb{R}^p$ (see [43, 68]). This shows that a smaller $\mu$ leads to a finer approximation.

The second approach for transforming the problem is the decomposition method in which we consider:

$$\min_{\boldsymbol{x}\in\mathbb{R}^n,\boldsymbol{z}\in\mathbb{R}^p}\left\{G_\rho(\boldsymbol{x},\boldsymbol{z}) = f(\boldsymbol{x}) + g(\boldsymbol{z}) + \frac{\rho}{2}\|\boldsymbol{z} - \boldsymbol{D}^*\boldsymbol{x}\|_2^2\right\}. \tag{4.17}$$

With $\rho \to \infty$, this problem is equivalent to the following constrained formulation of the original problem (4.13):

$$\begin{aligned} &\min\{f(\boldsymbol{x}) + g(\boldsymbol{z})\} \\ &\text{s.t.} \quad \boldsymbol{z} = \boldsymbol{D}^*\boldsymbol{x}, \quad \boldsymbol{x} \in \mathbb{R}^n, \boldsymbol{z} \in \mathbb{R}^p. \end{aligned} \tag{4.18}$$

Evidently, there is a close relationship between the approximate models (4.16) and (4.17). Indeed, fixing $\boldsymbol{x}$ and minimizing the objective function of (4.17) with respect to $\boldsymbol{z}$ we obtain

$$\min_{\boldsymbol{x}\in\mathbb{R}^n,\boldsymbol{z}\in\mathbb{R}^p}\left\{f(\boldsymbol{x}) + g(\boldsymbol{z}) + \frac{\rho}{2}\|\boldsymbol{z} - \boldsymbol{D}^*\boldsymbol{x}\|_2^2\right\} = \min_{\boldsymbol{x}\in\mathbb{R}^n}\left\{f(\boldsymbol{x}) + g_{\frac{1}{\rho}}(\boldsymbol{D}^*\boldsymbol{x})\right\}. \tag{4.19}$$

Therefore, the two models are equivalent in the sense that their optimal solution set (limited to $\boldsymbol{x}$) is the same when $\mu = \frac{1}{\rho}$. For analysis sparse recovery, both transformations lead to RALASSO (4.5). However, as we shall see, the resulting smoothing-based and decomposition-based algorithms and their analysis are very different.

### 4.2.4 The smoothing-based method

Since (4.16) is a smooth problem we can apply an optimal first-order method such as MFISTA with $F = H_\mu = f(\boldsymbol{x}) + g_\mu(\boldsymbol{D}^*\boldsymbol{x})$ and $G \equiv 0$ in equation (4.11). The Lipschitz constant of $H_\mu$ is given by $L_{\nabla f} + \frac{\|\boldsymbol{D}\|_2^2}{\mu}$, and according to (4.10) the gradient of $\nabla g_\mu(\boldsymbol{D}^*\boldsymbol{x})$ is equal to $\frac{1}{\mu}\boldsymbol{D}(\boldsymbol{D}^*\boldsymbol{x} - \mathrm{prox}_{\mu g}(\boldsymbol{D}^*\boldsymbol{x}))$. The expression $\mathrm{prox}_{\mu g}(\boldsymbol{D}^*\boldsymbol{x})$ is calculated by first computing $\mathrm{prox}_{\mu g}(\boldsymbol{z})$, and then letting $\boldsymbol{z} = \boldsymbol{D}^*\boldsymbol{x}$.

Returning to the analysis sparse recovery problem, after smoothing we obtain

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} \left\{ H_\mu(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|_2^2 + g_\mu(\boldsymbol{D}^*\boldsymbol{x}) \right\}, \tag{4.20}$$

where

$$\begin{aligned} g_\mu(\boldsymbol{D}^*\boldsymbol{x}) &= \min_{\boldsymbol{u}} \left\{ \lambda\|\boldsymbol{u}\|_1 + \frac{1}{2\mu}\|\boldsymbol{u} - \boldsymbol{D}^*\boldsymbol{x}\|_2^2 \right\} \\ &= \sum_{i=1}^{p} \lambda \mathcal{H}_{\lambda\mu}((\boldsymbol{D}^*\boldsymbol{x})[i]). \end{aligned}$$

The function $\mathcal{H}_\alpha(x)$ with parameter $\alpha > 0$ is the so-called Huber function [78], and is given by

$$\mathcal{H}_\alpha(x) = \begin{cases} \frac{1}{2\alpha}x^2 & \text{if } |x| < \alpha \\ |x| - \frac{\alpha}{2} & \text{otherwise.} \end{cases} \tag{4.21}$$

From (4.14), the gradient of $g_\mu(\boldsymbol{D}^*\boldsymbol{x})$ is equal to

$$\nabla g_\mu(\boldsymbol{D}^*\boldsymbol{x}) = \frac{1}{\mu}\boldsymbol{D}(\boldsymbol{D}^*\boldsymbol{x} - \Gamma_{\lambda\mu}(\boldsymbol{D}^*\boldsymbol{x})). \tag{4.22}$$

Applying MFISTA to (4.20), results in the SFISTA algorithm, summarized in Algorithm 1.

**Algorithm1: Smoothing-based MFISTA (SFISTA)**

**Input**: An upper bound $L \geq \|\boldsymbol{A}\|_2^2 + \frac{\|\boldsymbol{D}\|_2^2}{\mu}$.
**Step 0.** Take $\boldsymbol{y}_1 = \boldsymbol{x}_0, t_1 = 1$.
**Step k.** $(k \geq 1)$ Compute
$\nabla f(\boldsymbol{y}_k) = \boldsymbol{A}^*(\boldsymbol{A}\boldsymbol{y}_k - \boldsymbol{b})$.
$\nabla g_\mu(\boldsymbol{D}^*\boldsymbol{x}_{k-1}) = \frac{1}{\mu}\boldsymbol{D}(\boldsymbol{D}^*\boldsymbol{x}_{k-1} - \Gamma_{\lambda\mu}(\boldsymbol{D}^*\boldsymbol{x}_{k-1}))$.
$\boldsymbol{z}_k = \boldsymbol{y}_k - \frac{1}{L}(\nabla f(\boldsymbol{y}_k) + \nabla g_\mu(\boldsymbol{D}^*\boldsymbol{x}_{k-1}))$.
$t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2}$.
$\boldsymbol{x}_k = \operatorname{argmin}\{H_\mu(\boldsymbol{x}) : \boldsymbol{x} = \boldsymbol{z}_k, \boldsymbol{x}_{k-1}\}$.
$\boldsymbol{y}_{k+1} = \boldsymbol{x}_k + \frac{t_k}{t_{k+1}}(\boldsymbol{z}_k - \boldsymbol{x}_k) + \frac{t_k - 1}{t_{k+1}}(\boldsymbol{x}_k - \boldsymbol{x}_{k-1})$.

Table 4.3: Smoothing-based MFISTA

### 4.2.5 The decomposition-based method

We can also employ MFISTA on the decomposition model

$$\min_{\boldsymbol{x}\in\mathbb{R}^n, \boldsymbol{z}\in\mathbb{R}^p} \{G_\rho(\boldsymbol{x}, \boldsymbol{z}) = F_\rho(\boldsymbol{x}, \boldsymbol{z}) + G(\boldsymbol{x}, \boldsymbol{z})\}, \tag{4.23}$$

where we take the smooth part as $F_\rho(\boldsymbol{x}, \boldsymbol{z}) = f(\boldsymbol{x}) + \frac{\rho}{2}\|\boldsymbol{z} - \boldsymbol{D}^*\boldsymbol{x}\|_2^2$ and the nonsmooth part as $G(\boldsymbol{x}, \boldsymbol{z}) = g(\boldsymbol{z})$. In order to apply MFISTA to (4.17), we need to compute the proximal operator of $\alpha G$ for a given constant $\alpha > 0$, which is given by

$$\operatorname{prox}_{\alpha G}(\boldsymbol{x}, \boldsymbol{z}) = \begin{pmatrix} \boldsymbol{x} \\ \operatorname{prox}_{\alpha g}(\boldsymbol{z}) \end{pmatrix}. \tag{4.24}$$

In RALASSO (4.5), $G(\boldsymbol{x}, \boldsymbol{z}) = \lambda\|\boldsymbol{z}\|_1$ and $F_\rho(\boldsymbol{x}, \boldsymbol{z}) = \frac{1}{2}\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|_2^2 + \frac{1}{2}\rho\|\boldsymbol{z} - \boldsymbol{D}^*\boldsymbol{x}\|_2^2$. Therefore,

$$\operatorname{prox}_{\alpha G}(\boldsymbol{x}, \boldsymbol{z}) = \begin{pmatrix} \boldsymbol{x} \\ \Gamma_{\lambda\alpha}(\boldsymbol{z}) \end{pmatrix}. \tag{4.25}$$

The Lipschitz constant of $\nabla F$ is equal to $(\|\boldsymbol{A}\|_2^2 + \rho(1 + \|\boldsymbol{D}\|_2^2))$. By applying MFISTA directly, we have the DFISTA algorithm, stated in Algorithm 2.

---

**Algorithm 2: Decomposition-based MFISTA (DFISTA)**

---

**Input**: An upper bound $L \geq (\|\boldsymbol{A}\|_2^2 + \rho(1 + \|\boldsymbol{D}\|_2^2))$.
**Step 0.** Take $\boldsymbol{u}_1 = \boldsymbol{x}_0, \boldsymbol{v}_1 = \boldsymbol{z}_0, t_1 = 1$.
**Step k.** $(k \geq 1)$ Compute
$\nabla_{\boldsymbol{x}} F_\rho(\boldsymbol{u}_k, \boldsymbol{v}_k) = \boldsymbol{A}^*(\boldsymbol{A}\boldsymbol{u}_k - \boldsymbol{b}) + \rho\boldsymbol{D}(\boldsymbol{D}^*\boldsymbol{u}_k - \boldsymbol{v}_k)$.
$\nabla_{\boldsymbol{z}} F_\rho(\boldsymbol{u}_k, \boldsymbol{v}_k)) = \rho(\boldsymbol{v}_k - \boldsymbol{D}^*\boldsymbol{u}_k)$.
$\boldsymbol{p}_k = \boldsymbol{u}_k - \frac{1}{L}\nabla_{\boldsymbol{x}} F_\rho(\boldsymbol{u}_k, \boldsymbol{v}_k)$.
$\boldsymbol{q}_k = \Gamma_{\frac{\lambda}{L}}(\boldsymbol{v}_k - \frac{1}{L}\nabla_{\boldsymbol{z}} F_\rho(\boldsymbol{u}_k, \boldsymbol{v}_k))$.
$t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2}$.
$(\boldsymbol{x}_k, \boldsymbol{z}_k)$
$= \text{argmin}\{G_\rho(\boldsymbol{x}, \boldsymbol{z}) : (\boldsymbol{x}, \boldsymbol{z}) = (\boldsymbol{p}_k, \boldsymbol{q}_k), (\boldsymbol{x}_{k-1}, \boldsymbol{z}_{k-1})\}$.
$\boldsymbol{u}_{k+1} = \boldsymbol{x}_k + \frac{t_k}{t_{k+1}}(\boldsymbol{p}_k - \boldsymbol{x}_k) + \frac{t_k - 1}{t_{k+1}}(\boldsymbol{x}_k - \boldsymbol{x}_{k-1})$.
$\boldsymbol{v}_{k+1} = \boldsymbol{z}_k + \frac{t_k}{t_{k+1}}(\boldsymbol{q}_k - \boldsymbol{z}_k) + \frac{t_k - 1}{t_{k+1}}(\boldsymbol{z}_k - \boldsymbol{z}_{k-1})$.

---

Table 4.4: Decomposition-based MFISTA

## 4.3 Convergence analysis

In this section we analyze the convergence behavior of both the smoothing based and decomposition based methods. Convergence of smoothing algorithms has been treated in [68, 43]. In order to make the chapter self contained, we quote the main results here. We then analyze the convergence of the decomposition approach. Both methods require the same type of operations at each iteration: the computation of the gradient of the smooth function $f$, and of the proximal operator corresponding to $\alpha g$, which means that they have the same computational cost per iteration. However, we show that smoothing converges faster than decomposition based methods. Specifically, the smoothing-based algorithm is guaranteed to generate an $\varepsilon$-optimal solution within $O(1/\varepsilon)$ iterations, whereas the decomposition-based

approach requires $O(1/\varepsilon^{1.5})$ iterations. We prove the results by analyzing SFISTA and DFISTA for the general problem (4.13), however, the same analysis can be easily extended to other optimal first-order methods, such as the one described in [68].

### 4.3.1 Convergence of the smoothing-based method

For SFISTA the sequence $\{\boldsymbol{x}_k\}$ satisfies the following relationship [70]:

$$H_\mu(\boldsymbol{x}_k) - H_\mu(\hat{\boldsymbol{x}}_\mu) \leq \frac{2\left(L_{\nabla f} + \frac{\|\boldsymbol{D}\|_2^2}{\mu}\right)\Lambda_1}{(k+1)^2}, \tag{4.26}$$

where $\Lambda_1$ is an upper bound on the expression $\|\hat{\boldsymbol{x}}_\mu - \boldsymbol{x}_0\|_2$ with $\hat{\boldsymbol{x}}_\mu$ being an arbitrary optimal solution of the smoothed problem (4.16), and $\boldsymbol{x}_0$ is the initial point of the algorithm. Of course, this rate of convergence is problematic since we are more interested in bounding the expression $H(\boldsymbol{x}_k) - \hat{H}$ rather than the expression $H_\mu(\boldsymbol{x}_k) - H_\mu(\hat{\boldsymbol{x}}_\mu)$, which is in terms of the smoothed problem. Here, $\hat{H}$ stands for the optimal value for original nonsmooth problem (4.13). For that, we can use the following result from [43].

**Theorem 4.2.** *[43] Let $\{\boldsymbol{x}_k\}$ be the sequence generated by applying MFISTA to the problem (4.16). Let $\boldsymbol{x}_0$ be the initial point and let $\hat{\boldsymbol{x}}$ denote the optimal solution of (4.13). An $\varepsilon$-optimal solution of (4.13), i.e. $|H(\boldsymbol{x}_k) - H(\hat{\boldsymbol{x}})| \leq \varepsilon$, is obtained in the smoothing-based method using MFISTA after at most*

$$K = 2\|\boldsymbol{D}\|_2\sqrt{L_g\Lambda_1}\frac{1}{\varepsilon} + \sqrt{L_{\nabla f}\Lambda_1}\frac{1}{\sqrt{\varepsilon}} \tag{4.27}$$

*iterations with $\mu$ chosen as*

$$\mu = \sqrt{\frac{\|\boldsymbol{D}\|_2^2}{L_g}}\frac{\varepsilon}{\sqrt{\|\boldsymbol{D}\|_2^2 L_g} + \sqrt{\|\boldsymbol{D}\|_2^2 L_g + L_{\nabla f}\varepsilon}}, \tag{4.28}$$

*in which $L_g$ and $L_{\nabla f}$ are the Lipschitz constants of $g$ and the gradient function of $f$ in (4.13), and $\Lambda_1 = \|\boldsymbol{x}_0 - \hat{\boldsymbol{x}}_\mu\|_2$. We use $\hat{\boldsymbol{x}}_\mu$ to denote the optimal solution of problem (4.16).*

**Remarks:** For analysis sparse recovery using SFISTA, $L_g = \lambda p^{\frac{1}{2}}$ and $L_{\nabla f} = \|\boldsymbol{A}\|_2^2$, which can be plugged into the expressions in the theorem.

### 4.3.2 Convergence of the decomposition-based method

A key property of the decomposition model (4.17) is that its minimal value is bounded above by the optimal value $\hat{H}$ in the original problem (4.13).

**Lemma 4.1.** *Let $\hat{G}_\rho$ be the optimal value of problem* (4.17) *and $\hat{H}$ be the optimal value of problem* (4.13)*. Then $\hat{G}_\rho \leq \hat{H}$.*

**Proof:** The proof follows from adding the constraint $\boldsymbol{z} = \boldsymbol{D}^*\boldsymbol{x}$ to the optimization:

$$\begin{aligned}\hat{G}_\rho &= \min_{\boldsymbol{x}\in\mathbb{R}^n, \boldsymbol{z}\in\mathbb{R}^p} \left\{ f(\boldsymbol{x}) + g(\boldsymbol{z}) + \frac{\rho}{2}\|\boldsymbol{z} - \boldsymbol{D}^*\boldsymbol{x}\|_2^2 \right\} \\ &\leq \min_{\boldsymbol{x}\in\mathbb{R}^n, \boldsymbol{z}\in\mathbb{R}^p, \boldsymbol{z}=\boldsymbol{D}^*\boldsymbol{x}} \left\{ f(\boldsymbol{x}) + g(\boldsymbol{z}) + \frac{\rho}{2}\|\boldsymbol{z} - \boldsymbol{D}^*\boldsymbol{x}\|_2^2 \right\} \\ &= \min_{\boldsymbol{x}\in\mathbb{R}^n} \left\{ f(\boldsymbol{x}) + g(\boldsymbol{D}^*\boldsymbol{x}) \right\}, \end{aligned} \tag{4.29}$$

which is equal to $\hat{H}$.

The next theorem is our main convergence result establishing that an $\varepsilon$-optimal solution can be reached after $O(1/\varepsilon^{1.5})$ iterations. By assuming that the functions $f$ and $g$ are nonnegative, which is not an unusual assumption, we have the following theorem.

**Theorem 4.3.** *Let $\{\boldsymbol{x}_k, \boldsymbol{z}_k\}$ be the sequences generated by applying MFISTA to* (4.17) *with both $f$ and $g$ both being nonnegative functions. The initial point is taken as $(\boldsymbol{x}_0, \boldsymbol{z}_0)$ with $\boldsymbol{z}_0 = \boldsymbol{D}^*\boldsymbol{x}_0$. Let $\hat{\boldsymbol{x}}$ denote the optimal solution of the original problem* (4.13)*. An $\varepsilon$-optimal solution of problem* (4.13)*, i.e. $|H(\boldsymbol{x}_k) - H(\hat{\boldsymbol{x}})| \leq \varepsilon$, is obtained using the decomposition-based method after at most*

$$K = \max\left\{ \frac{16\sqrt{(1 + \|\boldsymbol{D}\|^2\Lambda_2 H(\boldsymbol{x}_0))}L_g}{\varepsilon^{1.5}}, \frac{2\sqrt{L_{\nabla f}\Lambda_2}}{\sqrt{\varepsilon}} \right\} \tag{4.30}$$

*iterations of MFISTA with $\rho$ chosen as*

$$\rho = \left( \frac{L_g\sqrt{2H(\boldsymbol{x}_0)}K^2}{2(1 + \|\boldsymbol{D}\|^2)\Lambda_2} \right)^{2/3}. \tag{4.31}$$

*Here $L_g$ and $L_{\nabla f}$ are the Lipschitz constants for $g$ and the gradient function of $f$ in* (4.13)*, and $\Lambda_2 = \|\boldsymbol{x}_0 - \hat{\boldsymbol{x}}_\rho\|_2^2 + \|\boldsymbol{z}_0 - \hat{\boldsymbol{z}}_\rho\|_2^2$. We use $\hat{\boldsymbol{x}}_\rho, \hat{\boldsymbol{z}}_\rho$ to denote the optimal solutions to* (4.17)*.*

**Proof:** Since the monotone version of FISTA is applied we have

$$\begin{aligned} & f(\boldsymbol{x}_k) + g(\boldsymbol{z}_k) + \frac{\rho}{2}\|\boldsymbol{z}_k - \boldsymbol{D}^*\boldsymbol{x}_k\|_2^2 \\ =& G_\rho(\boldsymbol{x}_k, \boldsymbol{z}_k) \leq G_\rho(\boldsymbol{x}_0, \boldsymbol{z}_0) = f(\boldsymbol{x}_0) + g(\boldsymbol{D}^*\boldsymbol{x}_0) = H(\boldsymbol{x}_0). \end{aligned} \tag{4.32}$$

With the assumption that $f$ and $g$ are nonnegative, it follows that

$$\frac{\rho}{2}\|\boldsymbol{z}_k - \boldsymbol{D}^*\boldsymbol{x}_k\|_2^2 \leq H(\boldsymbol{x}_0),$$

and therefore

$$\|\boldsymbol{z}_k - \boldsymbol{D}^*\boldsymbol{x}_k\|_2 \leq \sqrt{\frac{2H(\boldsymbol{x}_0)}{\rho}}. \tag{4.33}$$

The gradient of $f(\boldsymbol{x}) + \frac{\rho}{2}\|\boldsymbol{z} - \boldsymbol{D}^*\boldsymbol{x}\|_2^2$, is Lipschitz continuous with parameter $(L_{\nabla f} + \rho(1 + \|\boldsymbol{D}\|_2^2))$. According to [70], by applying MFISTA, we obtain a sequence $\{(\boldsymbol{x}_k, \boldsymbol{z}_k)\}$ satisfying

$$G_\rho(\boldsymbol{x}_k, \boldsymbol{z}_k) - \hat{G}_\rho \leq \frac{2(L_{\nabla f} + \rho(1 + \|\boldsymbol{D}\|_2^2))\Lambda_2}{k^2}.$$

Using lemma 4.1 and the notation

$$A = 2L_{\nabla f}\Lambda_2, B = 2(1 + \|\boldsymbol{D}\|_2^2)\Lambda_2,$$

we have

$$G_\rho(\boldsymbol{x}_k, \boldsymbol{z}_k) - \hat{H} \leq \frac{A + \rho B}{k^2}. \tag{4.34}$$

We therefore conclude that

$$\begin{aligned} H(\boldsymbol{x}_k) =& f(\boldsymbol{x}_k) + g(\boldsymbol{D}^*\boldsymbol{x}_k) \\ =& f(\boldsymbol{x}_k) + g(\boldsymbol{z}_k) + g(\boldsymbol{D}^*\boldsymbol{x}_k) - g(\boldsymbol{z}_k) \\ \leq& G_\rho(\boldsymbol{x}_k, \boldsymbol{z}_k) + L_g\|\boldsymbol{z}_k - \boldsymbol{D}^*\boldsymbol{x}_k\|_2 \\ \leq& \hat{H} + \frac{A+\rho B}{k^2} + L_g\|\boldsymbol{z}_k - \boldsymbol{D}^*\boldsymbol{x}_k\|_2 \\ \leq& \hat{H} + \frac{A+\rho B}{k^2} + L_g\sqrt{\frac{2H(\boldsymbol{x}_0)}{\rho}}. \end{aligned}$$

The first inequality follows from the Lipschitz condition for the function $g$, the second inequality is obtained from (4.34), and the last inequality is a result of (4.33).

We now seek the "best" $\rho$ that minimizes the upper bound, or equivalently, minimizes the term

$$\frac{A+\rho B}{k^2} + L_g\sqrt{\frac{2H(\boldsymbol{x}_0)}{\rho}} = \frac{A}{k^2} + C\rho + \frac{D}{\sqrt{\rho}}, \tag{4.35}$$

where $C = \frac{B}{k^2}$ and $D = L_g\sqrt{2H(\boldsymbol{x}_0)}$. Setting the derivative to zero, the optimal value of $\rho$ is $\rho = \left(\frac{D}{2C}\right)^{2/3}$, and

$$H(\boldsymbol{x}_k) \leq \hat{H} + \frac{A}{k^2} + 2C^{1/3}D^{2/3}. \tag{4.36}$$

Therefore, to obtain an $\varepsilon$-optimal solution, it is enough that

$$\frac{A}{k^2} \leq \frac{\varepsilon}{2}, \quad \frac{2B^{1/3}D^{2/3}}{k^{2/3}} \leq \frac{\varepsilon}{2}, \tag{4.37}$$

or

$$\begin{aligned} k \geq& \max\left\{\frac{4^{3/2}B^{1/2}D}{\varepsilon^{1.5}}, \frac{\sqrt{2A}}{\sqrt{\varepsilon}}\right\} \\ =& \max\left\{\frac{16\sqrt{(1+\|\boldsymbol{D}\|^2\Lambda_2 H(\boldsymbol{x}_0))}L_g}{\varepsilon^{1.5}}, \frac{2\sqrt{L_{\nabla f}\Lambda_2}}{\sqrt{\varepsilon}}\right\}, \end{aligned} \tag{4.38}$$

completing the proof.

**Remarks:**
1. As in SFISTA, when treating the analysis sparse recovery problem, $L_g = \lambda p^{\frac{1}{2}}$ and $L_{\nabla f} = \|\boldsymbol{A}\|_2^2$, which again can be plugged into the expressions in the theorem.
2. MFISTA is applied in SFISTA and DFISTA to guarantee a mathematical rigorous proof, i.e. the existence of equation (4.32). In real application, FISTA without monotone operations can also be applied to yield corresponding smoothing and decomposition based algorithms.

Comparing the results of smoothing-based and decomposition-based methods, we immediately conclude that the smoothing-based method is preferable. First, it requires only $O(1/\varepsilon)$ iterations to obtain an $\varepsilon$-optimal solution whereas the decomposition approach necessitates $O(1/\varepsilon^{3/2})$ iterations. Note that both bounds are better than the bound $O(1/\varepsilon^2)$ corresponding to general sub-gradient schemes for nonsmooth optimization. Second, the bound in the smoothing approach depends on $\sqrt{L_g}$, and not on $L_g$, as when using decomposition methods. This is important since, for example, when $g(\boldsymbol{z}) = \|\boldsymbol{z}\|_1$, we have $L_g = p^{\frac{1}{2}}$. In the smoothing approach the dependency on $p$ is of the form $p^{\frac{1}{4}}$ and not $p^{\frac{1}{2}}$, as when using the decomposition algorithm.

## 4.4 Performance bound

We now turn to analyze the recovery performance of analysis LASSO when smoothing and decomposition are applied. As we have seen, both transformations lead to the same RALASSO problem in (4.5). Our main result in this section shows that the reconstruction obtained by solving RALASSO is stable when $\boldsymbol{D}^*\boldsymbol{x}$ has rapidly decreasing coefficients and the noise in the model (4.1) is small enough. Our performance bound also depends on the choice of parameter $\rho$ in the objective function. Before stating the main theorems, we first introduce a definition and some useful lemmas, whose proofs are detailed in the Appendix D.

To ensure stable recovery, we require that the matrix $\boldsymbol{A}$ satisfies the D-RIP:

**Definition 4.1.** *(D-RIP) [24]. The measurement matrix $\boldsymbol{A}$ obeys the restricted isometry property adapted to $\boldsymbol{D}$ with constant $\sigma_s$ if*

$$(1-\sigma_s)\|\boldsymbol{v}\|_2^2 \leq \|\boldsymbol{A}\boldsymbol{v}\|_2^2 \leq (1+\sigma_s)\|\boldsymbol{v}\|_2^2 \tag{4.39}$$

*holds for all* $\boldsymbol{v} \in \Sigma_s = \{\boldsymbol{y} : \boldsymbol{y} = \boldsymbol{D}\boldsymbol{x} \text{ and } \|\boldsymbol{x}\|_0 \leq s\}$. *In other words,* $\Sigma_s$ *is the union of subspaces spanned by all subsets of* $s$ *columns of* $\boldsymbol{D}$.

The following lemma provides a useful inequality for matrices satisfying D-RIP.

**Lemma 4.2.** *Let* $\boldsymbol{A}$ *satisfy the D-RIP with parameter* $\sigma_{2s}$, *and assume that* $\boldsymbol{u}, \boldsymbol{v} \in \Sigma_s$. *Then,*

$$\mathrm{Re}\langle \boldsymbol{A}\boldsymbol{u}, \boldsymbol{A}\boldsymbol{v}\rangle \geq -\sigma_{2s}\|\boldsymbol{u}\|_2\|\boldsymbol{v}\|_2 + \mathrm{Re}\langle \boldsymbol{u}, \boldsymbol{v}\rangle. \tag{4.40}$$

In the following, $\hat{\boldsymbol{x}}_\rho$ denotes the optimal solution of RALASSO (4.5) and $\boldsymbol{x}$ is the original signal in the linear model (4.1); we also use $\boldsymbol{h}$ to represent the reconstruction error $\boldsymbol{h} = \hat{\boldsymbol{x}}_\rho - \boldsymbol{x}$. Let $\mathcal{T}$ be the indices of coefficients with $s$ largest magnitudes in the vector $\boldsymbol{D}^*\boldsymbol{x}$, and denote the complement of $\mathcal{T}$ by $\mathcal{T}^c$. Setting $\mathcal{T}_0 = \mathcal{T}$, we decompose $\mathcal{T}_0^c$ into sets of size $s$ where $\mathcal{T}_1$ denotes the locations of the $s$ largest coefficients in $\boldsymbol{D}^*_{\mathcal{T}^c}\boldsymbol{x}$, $\mathcal{T}_2$ denote the next $s$ largest coefficients and so on. Finally, we let $\mathcal{T}_{01} = \mathcal{T}_0 \cup \mathcal{T}_1$.

Using the result of Lemma 4.2 and the inequality $\|\boldsymbol{D}^*_{\mathcal{T}_0}\boldsymbol{h}\|_2 + \|\boldsymbol{D}^*_{\mathcal{T}_1}\boldsymbol{h}\|_2 \leq \sqrt{2}\|\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\|_2$ since $\mathcal{T}_0$ and $\mathcal{T}_1$ are disjoint, we have the following lemma.

**Lemma 4.3.** ***(D-RIP property)*** *Let* $\boldsymbol{h} = \hat{\boldsymbol{x}}_\rho - \boldsymbol{x}$ *be the reconstruction error in RALASSO* (4.5). *We assume that* $\boldsymbol{A}$ *satisfies the D-RIP with parameter* $\sigma_{2s}$ *and* $\boldsymbol{D}$ *is a tight frame. Then,*

$$\mathrm{Re}\langle \boldsymbol{A}\boldsymbol{h}, \boldsymbol{A}\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\rangle \geq (1-\sigma_{2s})\|\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\|_2^2 - \sqrt{2}s^{-\frac{1}{2}}\sigma_{2s}\|\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\|_2\|\boldsymbol{D}^*_{\mathcal{T}^c}\boldsymbol{h}\|_1. \tag{4.41}$$

Finally, the lemmas below show that the reconstruction error $\boldsymbol{h}$ and $\|\boldsymbol{D}^*_{\mathcal{T}^c}\boldsymbol{h}\|_1$ can not be very large.

**Lemma 4.4.** ***(Optimality condition)*** *The optimal solution* $\hat{\boldsymbol{x}}_\rho$ *for RALASSO* (4.5) *satisfies*

$$\|\boldsymbol{D}^*\boldsymbol{A}^*\boldsymbol{A}\boldsymbol{h}\|_\infty \leq \left(\frac{1}{2} + \|\boldsymbol{D}^*\boldsymbol{D}\|_{1,1}\right)\lambda. \tag{4.42}$$

**Lemma 4.5.** ***(Cone constraint)*** *The optimal solution $\hat{\boldsymbol{x}}_\rho$ for RALASSO (4.5) satisfies the following cone constraint,*

$$\|\boldsymbol{D}^*_{\mathcal{T}^c}\boldsymbol{h}\|_1 \leq \frac{\lambda}{\rho}p + 3\|\boldsymbol{D}^*_{\mathcal{T}}\boldsymbol{h}\|_1 + 4\|\boldsymbol{D}^*_{\mathcal{T}^c}\boldsymbol{x}\|_1. \tag{4.43}$$

We are now ready to state our main result.

**Theorem 4.4.** *Let $\boldsymbol{A}$ be an $m \times n$ measurement matrix, $\boldsymbol{D}$ an arbitrary $n \times p$ tight frame, and let $\boldsymbol{A}$ satisfy the D-RIP with $\sigma_{2s} < 0.1907$. Consider the measurement $\boldsymbol{b} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{w}$, where $\boldsymbol{w}$ is noise that satisfies $\|\boldsymbol{D}^*\boldsymbol{A}^*\boldsymbol{w}\|_\infty \leq \frac{\lambda}{2}$. Then the solution $\hat{\boldsymbol{x}}_\rho$ to RALASSO (4.5) satisfies*

$$\|\hat{\boldsymbol{x}}_\rho - \boldsymbol{x}\|_2 \leq C_0\sqrt{s}\lambda + C_1\frac{\|\boldsymbol{D}^*\boldsymbol{x} - (\boldsymbol{D}^*\boldsymbol{x})_s\|_1}{\sqrt{s}} + C_2\frac{\lambda p}{\sqrt{s}\rho}, \tag{4.44}$$

*for the decomposition transformation and*

$$\|\hat{\boldsymbol{x}}_\rho - \boldsymbol{x}\|_2 \leq C_0\sqrt{s}\lambda + C_1\frac{\|\boldsymbol{D}^*\boldsymbol{x} - (\boldsymbol{D}^*\boldsymbol{x})_s\|_1}{\sqrt{s}} + C_2\frac{\lambda\mu p}{\sqrt{s}}, \tag{4.45}$$

*for the smoothing transformation. Here $(\boldsymbol{D}^*\boldsymbol{x})_s$ is the vector consisting of the largest $s$ entries of $\boldsymbol{D}^*\boldsymbol{x}$ in magnitude, $C_1$ and $C_2$ are constants depending on $\sigma_{2s}$, and $C_0$ depends on $\sigma_{2s}$ and $\|\boldsymbol{D}^*\boldsymbol{D}\|_{1,1}$.*

**Proof:** The proof follows mainly from the ideas in [40], [73], and proceeds in two steps. First, we try to show that $\boldsymbol{D}^*\boldsymbol{h}$ inside $\mathcal{T}_{01}$ is bounded by the terms of $\boldsymbol{D}^*\boldsymbol{h}$ outside the set $\mathcal{T}$. Then we show that $\boldsymbol{D}^*_{\mathcal{T}^c}\boldsymbol{h}$ is essentially small.

From Lemma 4.3,

$$\mathrm{Re}\langle \boldsymbol{A}\boldsymbol{h}, \boldsymbol{A}\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\rangle \geq (1-\sigma_{2s})\|\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\|_2^2 - \sqrt{2}s^{-\frac{1}{2}}\sigma_{2s}\|\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\|_2\|\boldsymbol{D}^*_{\mathcal{T}^c}\boldsymbol{h}\|_1. \tag{4.46}$$

Using the fact that $\mathrm{Re}\langle \boldsymbol{x}, \boldsymbol{y}\rangle \leq |\langle \boldsymbol{x}, \boldsymbol{y}\rangle| \leq \|\boldsymbol{x}\|_1\|\boldsymbol{y}\|_\infty$, we obtain that

$$\begin{aligned}
\mathrm{Re}\langle \boldsymbol{A}\boldsymbol{h}, \boldsymbol{A}\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\rangle =& \mathrm{Re}\langle \boldsymbol{D}^*\boldsymbol{A}^*\boldsymbol{A}\boldsymbol{h}, \boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\rangle \\
\leq& \|\boldsymbol{D}^*\boldsymbol{A}^*\boldsymbol{A}\boldsymbol{h}\|_\infty\|\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\|_1 \\
\leq& \sqrt{2s}c_0\lambda\|\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\|_2,
\end{aligned} \tag{4.47}$$

with $c_0 = \frac{1}{2} + \|\boldsymbol{D}^*\boldsymbol{D}\|_{1,1}$. The second inequality is a result of Lemma 4.4 and the fact that $\|\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\|_1 \leq \sqrt{2s}\|\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\|_2$, in which $2s$ is the number of nonzero terms in $\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}$. Combining (4.46) and (4.47), we get

$$\|\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\|_2 \leq \frac{\sqrt{2s}\lambda c_0 + \sqrt{2}s^{-\frac{1}{2}}\sigma_{2s}\|\boldsymbol{D}^*_{\mathcal{T}^c}\boldsymbol{h}\|_1}{1-\sigma_{2s}}. \tag{4.48}$$

Then the second step bounds $\|\boldsymbol{D}^*_{\mathcal{T}^c}\boldsymbol{h}\|_1$. From (4.48),

$$\begin{aligned}\|\boldsymbol{D}^*_{\mathcal{T}}\boldsymbol{h}\|_1 \leq& \sqrt{s}\|\boldsymbol{D}^*_{\mathcal{T}}\boldsymbol{h}\|_2 \leq \sqrt{s}\|\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\|_2 \\ \leq& \frac{\sqrt{2}\lambda s c_0 + \sqrt{2}\sigma_s\|\boldsymbol{D}^*_{\mathcal{T}^c}\boldsymbol{h}\|_1}{1-\sigma_{2s}}.\end{aligned} \tag{4.49}$$

Finally, using Lemma 4.5 and (4.49),

$$\|\boldsymbol{D}^*_{\mathcal{T}^c}\boldsymbol{h}\|_1 \leq \frac{\lambda}{\rho}p + \frac{3\sqrt{2}\lambda s c_0 + 3\sqrt{2}\sigma_{2s}\|\boldsymbol{D}^*_{\mathcal{T}^c}\boldsymbol{h}\|_1}{1-\sigma_{2s}} + 4\|\boldsymbol{D}^*_{\mathcal{T}^c}\boldsymbol{x}\|_1. \tag{4.50}$$

Since $\sigma_{2s} < 0.1907$, we have $1-(1+3\sqrt{2})\sigma_{2s} > 0$. Rearranging terms, the above inequality becomes

$$\|\boldsymbol{D}^*_{\mathcal{T}^c}\boldsymbol{h}\|_1 \leq \frac{1-\sigma_{2s}}{1-(1+3\sqrt{2})\sigma_{2s}}\frac{\lambda}{\rho}p + \frac{3\sqrt{2}\lambda s c_0 + 4(1-\sigma_{2s})\|\boldsymbol{D}^*_{\mathcal{T}^c}\boldsymbol{x}\|_1}{1-(1+3\sqrt{2})\sigma_{2s}}. \tag{4.51}$$

We now derive the bound on the reconstruction error. Using the results of (4.48) and (4.51), we get

$$\begin{aligned}\|\boldsymbol{h}\|_2 =& \|\boldsymbol{D}^*\boldsymbol{h}\|_2 \leq \|\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\|_2 + \sum_{j\geq 2}\|\boldsymbol{D}^*_{\mathcal{T}_j}\boldsymbol{h}\|_2 \\ \leq& \frac{\sqrt{2s}\lambda c_0 + \sqrt{2}s^{-\frac{1}{2}}\sigma_{2s}\|\boldsymbol{D}^*_{\mathcal{T}^c}\boldsymbol{h}\|_1}{1-\sigma_{2s}} + s^{-\frac{1}{2}}\|\boldsymbol{D}^*_{\mathcal{T}^c}\boldsymbol{h}\|_1 \\ =& \frac{c_0\lambda\sqrt{2s}}{1-\sigma_{2s}} + \frac{((\sqrt{2}-1)\sigma_{2s}+1)s^{-\frac{1}{2}}\|\boldsymbol{D}^*_{\mathcal{T}^c}\boldsymbol{h}\|_1}{1-\sigma_{2s}} \\ \leq& C_0\sqrt{s}\lambda + C_1\frac{\|\boldsymbol{D}^*\boldsymbol{x} - (\boldsymbol{D}^*\boldsymbol{x})_s\|_1}{\sqrt{s}} + C_2\frac{\lambda p}{\sqrt{s}\rho}.\end{aligned} \tag{4.52}$$

The first equality follows from the assumption that $\boldsymbol{D}$ is a tight frame so that $\boldsymbol{D}\boldsymbol{D}^* = \boldsymbol{I}$. The first inequality is the result of the triangle inequality. The second inequality follows from (4.48) and the fact that $\sum_{j\geq 2} \|\boldsymbol{D}^*_{\mathcal{T}_j}\boldsymbol{h}\|_2 \leq s^{-\frac{1}{2}}\|\boldsymbol{D}^*_{\mathcal{T}^c}\boldsymbol{h}\|_1$, which is proved in equation (D.1) in the Appendix D. The constants in the final result are given by

$$\begin{aligned} C_0 =& \frac{4\sqrt{2}c_0}{1-(1+3\sqrt{2})\sigma_{2s}}, \\ C_1 =& \frac{4((\sqrt{2}-1)\sigma_{2s}+1)}{1-(1+3\sqrt{2})\sigma_{2s}}, \\ C_2 =& \frac{(\sqrt{2}-1)\sigma_{2s}+1}{1-(1+3\sqrt{2})\sigma_{2s}}. \end{aligned}$$

To obtain the error bound for the smoothing transformation we replace $\rho$ with $1/\mu$ in the result. $\square$

Choosing $\rho \to \infty$ in RALASSO (4.5) leads to the ALASSO problem for which $\boldsymbol{z} = \boldsymbol{D}^*\boldsymbol{x}$. We then have the following result.

**Theorem 4.5.** *Let $\boldsymbol{A}$ be an $m \times n$ measurement matrix, $\boldsymbol{D}$ an arbitrary $n \times p$ tight frame, and let $\boldsymbol{A}$ satisfy the D-RIP with $\sigma_{2s} < 0.1907$. Consider the measurement $\boldsymbol{b} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{w}$, where $\boldsymbol{w}$ is noise that satisfies $\|\boldsymbol{D}^*\boldsymbol{A}^*\boldsymbol{w}\|_\infty \leq \frac{\lambda}{2}$. Then the solution $\hat{\boldsymbol{x}}$ to ALASSO (4.4) satisfies*

$$\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2 \leq C_0\sqrt{s}\lambda + C_1\frac{\|\boldsymbol{D}^*\boldsymbol{x} - (\boldsymbol{D}^*\boldsymbol{x})_s\|_1}{\sqrt{s}}, \tag{4.53}$$

*where $(\boldsymbol{D}^*\boldsymbol{x})_s$ is the vector consisting of the largest $s$ entries of $\boldsymbol{D}^*\boldsymbol{x}$ in magnitude, $C_1$ is a constant depending on $\sigma_{2s}$, and $C_0$ depends on $\sigma_{2s}$ and $\|\boldsymbol{D}^*\boldsymbol{D}\|_{1,1}$.*

**Remarks:**

1. When the noise in the system is zero, we can set $\lambda$ as a positive value which is arbitrarily close to zero. The solution $\hat{\boldsymbol{x}}$ then satisfies $\|\hat{\boldsymbol{x}} - \boldsymbol{x}\| \leq C_1\frac{\|\boldsymbol{D}^*\boldsymbol{x} - (\boldsymbol{D}^*\boldsymbol{x})_s\|_1}{\sqrt{s}}$, which parallels the result for the noiseless synthesis model in [40].

2. When $\boldsymbol{D}^*$ is a tight frame, we have $\boldsymbol{D}\boldsymbol{D}^* = \boldsymbol{I}$. Therefore by letting $\boldsymbol{v} = \boldsymbol{D}^*\boldsymbol{x}$, we can reformulate the original analysis model as

$$\min_{\boldsymbol{v}} \frac{1}{2}\|\boldsymbol{A}\boldsymbol{D}\boldsymbol{v} - \boldsymbol{b}\|_2^2 + \lambda\|\boldsymbol{v}\|_1. \tag{4.54}$$

Assuming that the noise term satisfies the $l_2$ norm constraint $\|\boldsymbol{w}\|_2 \leq \varepsilon$, we have

$$\|\boldsymbol{D}^*\boldsymbol{A}^*\boldsymbol{w}\|_\infty \leq \|\boldsymbol{D}^*\boldsymbol{A}^*\boldsymbol{w}\|_2 \leq \|\boldsymbol{D}^*\boldsymbol{A}^*\|_2\|\boldsymbol{w}\|_2 \leq \varepsilon\|\boldsymbol{D}^*\boldsymbol{A}^*\|_2. \tag{4.55}$$

When $\boldsymbol{A}$ satisfies D-RIP with $\sigma_{2s} < 0.1907$, by letting $\lambda = 2\varepsilon\|\boldsymbol{D}^*\boldsymbol{A}^*\|_2$ we have

$$\|\hat{\boldsymbol{v}} - \boldsymbol{v}\|_2 \leq \|\boldsymbol{D}^*\|_2\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2 \leq \tilde{C}_0\varepsilon + \tilde{C}_1\frac{\|\boldsymbol{v} - (\boldsymbol{v})_s\|_1}{\sqrt{s}}. \tag{4.56}$$

This result has a form similar to the reconstruction error bound shown in [40]. However, the specific constants are different since in [40] the matrix $\boldsymbol{A}\boldsymbol{D}$ is required to satisfy the RIP, whereas in our work we require only that the D-RIP is satisfied.

3. A similar performance bound is introduced in [73] and shown to be valid when $\sigma_{3s} < 0.25$. Using Corollary 3.4 in [79], this is equivalent to $\sigma_{2s} < 0.0833$. Thus the results in Theorem 4.5 allow for a looser constraint on ALASSO recovery.

4. The performance bound of Theorem 4.4 implies that a larger choice of $\rho$, or a smaller parameter $\mu$, leads to a smaller reconstruction error bound. This trend is intuitive since large $\rho$ or small $\mu$ results in smaller model inaccuracy. However, a larger $\rho$ or a smaller $\mu$ leads to a larger Lipschitz constant and thus results in slower convergence according to Theorem 4.1. The idea of parameter continuation [44] can be introduced to both $\rho$ and $\mu$ to accelerate the convergence while obtaining a desired reconstruction accuracy. More details will be given in the next section.

## 4.5 Numerical examples

In the numerical examples, we use both randomly generated data and MRI image reconstruction to demonstrate that SFISTA performs better than DFISTA. In the last example we also

introduce a continuation technique to further speed up convergence of the smoothing-based method. We further compare SFISTA with the existing methods in [66, 67, 69] using MRI image reconstruction, and show its advantages.

### 4.5.1 Randomly generated data in a noiseless case

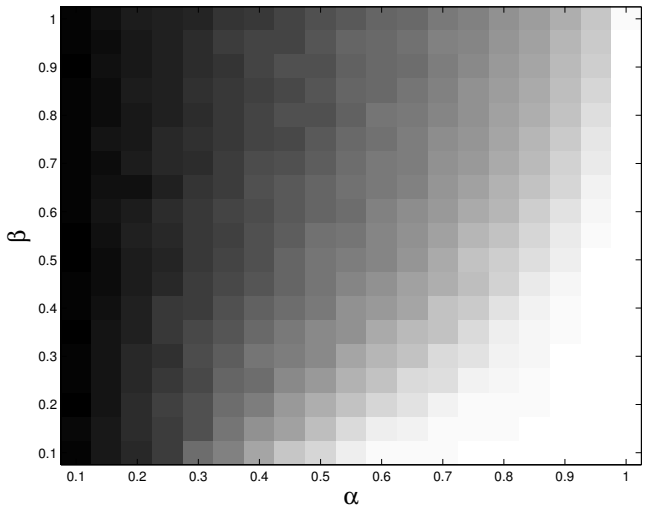


Figure 4.1: Reconstruction error of SFISTA

In this simulation, the entries in the $m \times n$ measurement matrix $\boldsymbol{A}$ were randomly generated according to a normal distribution. The $n \times p$ matrix $\boldsymbol{D}$ is a random tight frame. First we generated a $p \times n$ matrix whose elements follow an i.i.d Gaussian distribution. Then QR factorization was performed on this random matrix to yield the tight frame $\boldsymbol{D}$ with $\boldsymbol{D}\boldsymbol{D}^* = \boldsymbol{I}$ ($\boldsymbol{D}^*$ comprises the first $n$ columns from $\boldsymbol{Q}$, which was generated from the QR factorization).

In the simulation we let $n = 120$ and $p = 144$, and we also set the values of $m$ and the number of zero terms named $l$ in $\boldsymbol{D}^*\boldsymbol{x}$ according to the following formula:

$$m = \alpha n, \quad l = n - \beta m. \tag{4.57}$$

We varied $\alpha$ and $\beta$ from 0.1 to 1, with a step size 0.05. We set $\lambda = 0.004$, $\mu = 10^{-3}\lambda^{-1}$ for the smoothing-based method, and $\rho = 10^3\lambda$ for the decomposition-based method. $L$ is

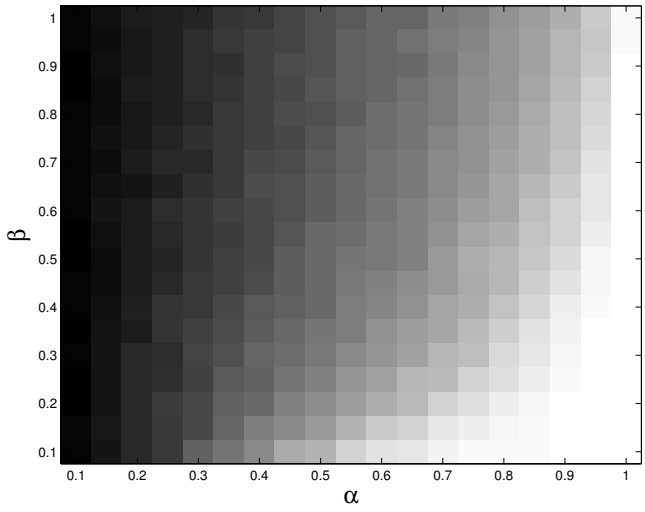


Figure 4.2: Reconstruction error of DFISTA

set to be $\|\boldsymbol{A}\|_2^2 + \frac{\|\boldsymbol{D}\|_2^2}{\mu}$ for smoothing and $\|\boldsymbol{A}\|_2^2 + \rho(1 + \|\boldsymbol{D}\|_2^2)$ for decomposition. For every combination of $\alpha$ and $\beta$, we ran a Monte Carlo simulation 50 times. Each algorithm ran for 3000 iterations, and we computed the average reconstruction error. The reconstruction error is defined by $\frac{\|\hat{\boldsymbol{x}}-\boldsymbol{x}\|}{\|\boldsymbol{x}\|}$, in which $\hat{\boldsymbol{x}}$ is the reconstructed signal using smoothing or decomposition and $\boldsymbol{x}$ is the original signal in (4.1).

The average reconstruction error for smoothing and decomposition are plotted in Figs. 4.1 and 4.2, respectively. White pixels present low reconstruction error whereas black pixels mean high error. Evidently, see that with same number of iterations, SFISTA results in a better reconstruction than DFISTA.

### 4.5.2 MRI image reconstruction in a noisy case

The next numerical experiment was performed on a noisy $256 \times 256$ Shepp Logan phantom. The image scale was normalized to $[0, 1]$. The additive noise followed a zero-mean Gaussian distribution with standard deviation $\sigma = 0.001$. Due to the high cost of sampling in MRI, we only observed a limited number of radial lines of the phantom's 2D discrete Fourier transform. The matrix $\boldsymbol{D}^*$ consists of all vertical and horizontal gradients, which leads to

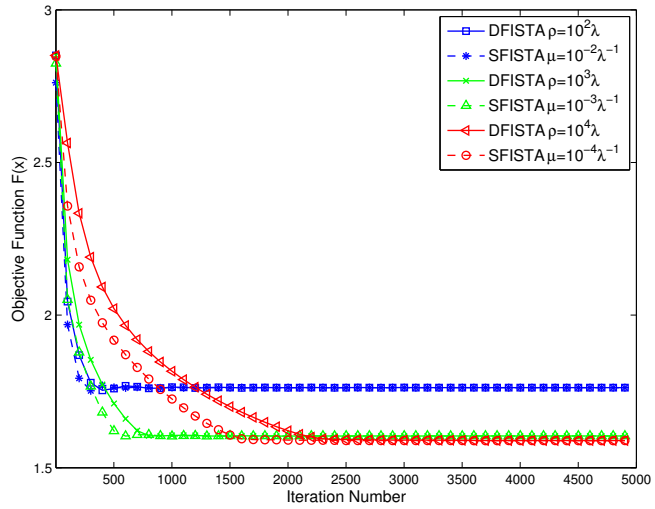


Figure 4.3: The objective function for MRI reconstruction on Shepp Logan.

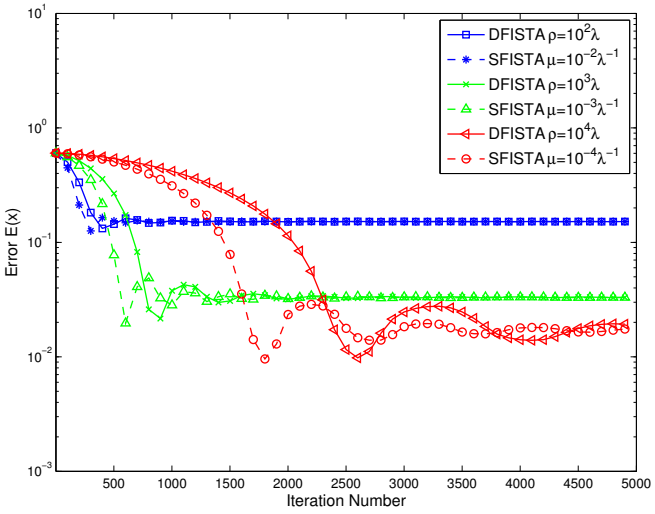


Figure 4.4: Reconstruction error for SFISTA and DFISTA with different parameters.

a sparse $\boldsymbol{D}^*\boldsymbol{x}$. We let $\lambda = 0.001$ in the optimization. We tested this MRI scenario with $\mu$ values of $10^{-2}\lambda^{-1}, 10^{-3}\lambda^{-1}, 10^{-4}\lambda^{-1}$ for SFISTA and $\rho = 10^2\lambda, \rho = 10^3\lambda, 10^4\lambda$ for DFISTA. $L$ is set to be $\|\boldsymbol{A}\|_2^2 + \frac{\|\boldsymbol{D}\|_2^2}{\mu}$ for SFISTA and $\|\boldsymbol{A}\|_2^2 + \rho(1 + \|\boldsymbol{D}\|_2^2)$ for DFISTA. We took the samples along 15 radial lines to test these two methods.

In Fig. 4.3 we plot the objective $\frac{1}{2}\|\boldsymbol{A}\boldsymbol{x}-\boldsymbol{b}\|_2^2+\lambda\|\boldsymbol{D}^*\boldsymbol{x}\|_1$ as a function of the iteration number. It can be seen that the objective function of SFISTA decreases more rapidly than DFISTA. Furthermore, with smaller $\rho$ and larger $\mu$, DFISTA and SFISTA converge faster. Then we computed the reconstruction error. Here we see that smaller $\mu$ and larger $\rho$ lead to a more accurate reconstruction. We can see that SFISTA converges faster than DFISTA, which follows the convergence results in Section 4.3.

Next, we compared SFISTA with the nonlinear conjugate gradient descend (CGD) algorithm proposed in [69]. The CGD also needs to introduce a smoothing transformation to approximate the term $\|\boldsymbol{D}^*\boldsymbol{x}\|_1$, and in this simulation the Moreau envelop with $\mu=10^{-4}\lambda^{-1}$ was used to smooth this term. We can see from Fig. 4.5 that SFISTA converges faster than the CGD in terms of CPU time. CGD is slower because in each iteration, backtracking line-search is required, which reduces the algorithm efficiency.

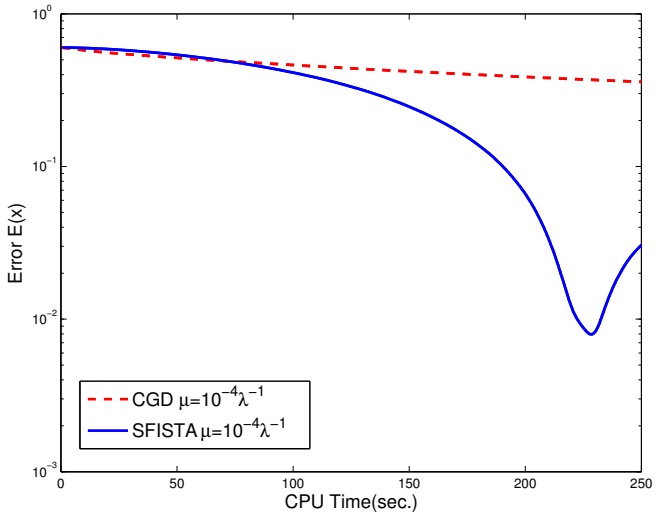


Figure 4.5: Reconstruction error for SFISTA and CGD with respect to CPU time.

### 4.5.3 Acceleration by continuation

To accelerate convergence and increase the accuracy of reconstruction, we consider continuation on the parameter $\mu$ for SFISTA, or on $\rho$ for DFISTA. From Theorem 4.4, we see

**Algorithm 3: Continuation with SFISTA**

**Input**: $\boldsymbol{x}$, the starting parameter $\mu = \mu_0$,
the ending parameter $\mu_f$ and $\gamma > 1$.
**Step 1.** run SFISTA with $\mu$ and initial point $\boldsymbol{x}$.
**Step 2.** Get the solution $\boldsymbol{x}^*$ and let $\boldsymbol{x} = \boldsymbol{x}^*, \mu = \mu/\gamma$.
**Until.** $\mu \leq \mu_f$.

Table 4.5: Continuation with SFISTA

that smaller $\mu$ results in a smaller reconstruction error. At the same time, smaller $\mu$ leads to a larger Lipschitz constant $L_{\nabla F}$ in Theorem 4.1, and thus results in slower convergence. The idea of continuation is to solve a sequence of similar problems while using the previous solution as a warm start. Taking the smoothing-based method as an example, we can run SFISTA with $\mu_1 \geq \mu_2 \geq \mu_3, \ldots \geq \mu_f$. The continuation method is given in Algorithm 3. The algorithm for applying continuation on DFISTA is the same.

We tested the algorithm on the Shepp Logan image from the previous subsection with the same setting, using SFISTA with $\mu_f = 10^{-4}\lambda^{-1}$ and standard SFISTA with $\mu = 10^{-4}\lambda^{-1}$. We implemented the generalized iterative soft-thresholding algorithm (GIST) from [67]. We also included an ADMM-based method, i.e. the split augmented Lagrangian shrinkage algorithm (SALSA) [66]. SALSA requires solving the proximal operator of $\|\boldsymbol{D}^*\boldsymbol{x}\|_1$, which is nontrivial. In this simulation, we implemented 40 iterations of the Fast GP algorithm [70] to approximate this proximal operator. Without solving the proximal operator exactly, the ADMM-based method can converge very fast while the accuracy of reconstruction is compromised as we show in Figure 4.6. In this figure we plot the reconstruction error for these four algorithms. It also shows that continuation helps speed up the convergence and exhibits better performance then GIST. The reconstructed Shepp Logan phantom using continuation is presented in Fig. 4.7, with reconstruction error 3.17%.

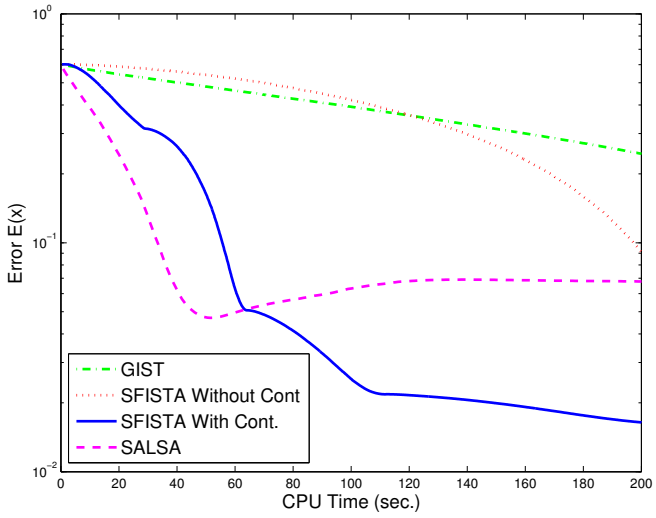


Figure 4.6: Convergence comparison among SFISTA with and without continuation, GIST and SALSA.

Figure 4.7: Reconstructed Shepp Logan with SFISTA using continuation.

## 4.6 Summary

In this chapter we considered the analysis model of sparse recovery. Efficient algorithms based on MFISTA were derived to solve the LASSO optimization for the analysis model.

Since the proximal operator in MFISTA does not have a closed-form solution for the analysis model, we introduced two relaxation methods, namely smoothing and decomposition, to transform the original sparse recovery problem into a smooth counterpart in which a proximal operator exists. Although smoothing and decomposition relaxations have the same objective function, by performing convergence analysis we showed that the smoothing method converges faster than the decomposition method. We also derived a bound on the performance for both approaches, assuming a tight frame and D-RIP. We demonstrated the effectiveness of the proposed algorithms for an MRI application. With the application of parameter continuation, the proposed algorithms are suitable for solving other large scale problems.

# Chapter 5

# Conclusions and Future Work

## 5.1 Summary and conclusions

In this dissertation we studied several topics related to sparse recovery and their applications to radar signal processing, passive arrays, and image processing. We first proposed a method to overcome structured dictionary mismatches in compressed sensing. In particular, we utilized the joint sparse recovery model and also derived a performance bound on the joint sparse reconstruction. For off-grid compressed sensing, we used a bounded joint sparse recovery method. Fast algorithms based on FISTA were derived to solve these joint sparse recovery formulations. One important application of this framework is called off-grid compressed sensing for DOA estimation. We presented both passive and active sensing applications to demonstrate the effectiveness of the proposed algorithms. Numerical examples were conducted to compare the performance of the joint sparse method and other existing methods. By exploiting the joint sparse property, we achieved considerably more satisfactory results when structured dictionary mismatches exist.

We next extended the sparse recovery to a continuous domain where dictionary mismatches do not exist. In particular, we extended the mathematical theory of super resolution to DOA estimation using co-prime arrays. A primal-dual approach was utilized to transform the original infinite dimensional optimization problem to a solvable semidefinite program. After estimating the candidate support sets by root finding, we solved a small-scale sparse recovery problem. The robustness of the proposed super resolution approach was verified by performing statistical analysis of the noise inherent in co-prime array processing. A source-number detection algorithm was then proposed by combining the existing SORTE algorithm

with the reconstructed spectrum from convex optimization. Via numerical examples, we showed that the proposed method achieves a more accurate DOA estimation while providing more degrees of freedom, and also exhibits improved resolution over traditional MUSIC with spatial smoothing.

Finally, we considered the problem of analysis model for sparse recovery. We proposed methods based on MFISTA to solve the LASSO optimization problem for the analysis model. Since the proximal operator in MFISTA for $\|\boldsymbol{D}^*\boldsymbol{x}\|_1$ does not have a closed-form solution, we presented two methods, SFISTA and DFISTA, using smoothing and decomposition respectively, to transform the original sparse recovery problem into a smooth counterpart. We analyzed the convergence of SFISTA and DFISTA and showed that SFISTA converges faster for general nonsmooth optimization problems. We also derived a bound on the performance for both approaches, assuming a tight frame and D-RIP. Our methods were demonstrated via several simulations. With the application of parameter continuation, these two algorithms are suitable for solving large scale problems.

## 5.2 Future directions

In this section, we point out several potential future research directions.

**Performance analysis of the DOA estimation accuracy:** In the dissertation we focused on the estimation accuracy of the original sparse signal when off-grid targets exist. However, the DOA estimation accuracy is a more interesting topic to explore. This performance can be analyzed with respect to the restricted isometric property of the sensing matrix. A more theoretical way to choose $\lambda$ for off-grid compressed sensing can also developed.

**Efficient algorithms to solve continuous sparse recovery:** Although continuous sparse recovery method requires less sampling time than MUSIC to achieve a certain estimation accuracy, one shortcoming of this approach is that solving the semidefinite program is more time consuming than MUSIC. Fast algorithm development could be an interesting topic for future work. It would also be of interest to develop a systematic way to choose $\epsilon$ and $\epsilon_d$ in the optimization formulas.

**Random arrays for passive DOA estimation:** In this dissertation we consider only nested arrays and co-prime arrays to increase the degrees of freedom from $O(M+N)$ to $O(MN)$. In future work, we will consider the case of random arrays, where locations of sensors are randomly generated. The challenge is the distribution of the random generation. Under proper distribution, it can be shown that random arrays can yield more degrees of freedom than nested and co-prime arrays using the same number of sensors.

**Dictionary learning of the analysis sparsity model:** The dissertation considers only recovery of the co-sparse signal with a given dictionary. However, in most applications, we need not only to estimate the signal but also to learn the overcomplete dictionary with the training set, which makes the problem itself nonconvex and hard to solve. The issue of reconstructing the dictionary can be addressed by performing a clustering analysis on the samples. Furthermore, co-sparse dictionary learning can be incorporated with machine learning techniques, such as the Support Vector Machines, to utilize the low dimensional structure of the data, thus increasing the discriminative power of original machine learning techniques.

# References


[1] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 21 –30, Mar. 2008.

[2] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 489 – 509, Feb. 2006.

[3] E. J. Candès and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?" *IEEE Trans. Inf. Theory*, vol. 52, no. 12, pp. 5406 –5425, Dec. 2006.

[4] Y. C. Eldar and G. Kutyniok, *Compressed Sensing: Theory and Applications*. Cambridge University Press, 2012.

[5] J. Romberg, "Imaging via compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 14 –20, Mar. 2008.

[6] T. Strohmer and B. Friedlander, "Compressed sensing for MIMO radar - algorithms and performance," in *Signals, Systems and Computers, 2009 Conference Record of the Forty-Third Asilomar Conference on*, Nov. 2009, pp. 464 –468.

[7] Y. Yu, A. P. Petropulu, and H. V. Poor, "MIMO radar using compressive sampling," *IEEE J. Sel. Topics Signal Process.*, vol. 4, no. 1, pp. 146 –163, Feb. 2010.

[8] S. Gogineni and A. Nehorai, "Target estimation using sparse modeling for distributed MIMO radar," *IEEE Trans. Signal Process.*, vol. 59, no. 11, pp. 5315 –5325, Nov. 2011.

[9] W. Bajwa, J. Haupt, A. Sayeed, and R. Nowak, "Compressive wireless sensing," in *Information Processing in Sensor Networks, 2006. IPSN 2006. The Fifth International Conference on*, 2006, pp. 134 –142.

[10] T. Blumensath and M. E. Davies, "Iterative hard thresholding for compressed sensing," *Applied and Computational Harmonic Analysis*, vol. 27, no. 3, pp. 265 – 274, 2009. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1063520309000384

[11] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. Inf. Theory*, vol. 53, no. 12, pp. 4655 –4666, Dec. 2007.

[12] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Rev.*, vol. 43, no. 3, pp. 129 –159, Mar. 2001.

[13] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 58, no. 1, pp. pp. 267–288, 1996. [Online]. Available: http://www.jstor.org/stable/2346178

[14] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM. on Imaging Sciences 2(1)*, pp. 183–202, Jan 2009.

[15] S. Becker, J. Bobin, and E. J. Candès, "Nesta: a fast and accurate first-order method for sparse recovery," *SIAM J. on Imaging Sciences 4(1)*, pp. 1–39.

[16] M. E. Tipping, "Sparse bayesian learning and the relevance vector machine," *Journal of Machine Learning Research*, vol. 1, pp. 211–244, 2001.

[17] S. Ji, Y. Xue, and L. Carin, "Bayesian compressive sensing," *IEEE Trans. Signal Process.*, vol. 56, no. 6, pp. 2346–2356, June 2008.

[18] Y. Chi, L. L. Scharf, A. Pezeshki, and A. R. Calderbank, "Sensitivity to basis mismatch in compressed sensing," *IEEE Trans. Signal Process.*, vol. 59, no. 5, pp. 2182 –2195, May 2011.

[19] M. Rosenbaum and A. B. Tsybakov, "Spase recovery under matrix uncertainty," *Annals of Statistics*, 2010.

[20] Z. Yang, C. Zhang, and L. Xie, "Robustly stable signal recovery in compressed sensing with structured matrix perturbation," *IEEE Trans. Signal Process.*, vol. 60, no. 9, pp. 4658–4671, 2012.

[21] H. Zhu, G. Leus, and G. B. Giannakis, "Sparsity-cognizant total least-squares for perturbed compressive sampling," *IEEE Trans. Signal Process.*, May 2011.

[22] E. J. Candès and C. Fernandez-Granda, "Towards a mathematical theory of super-resolution," *Communications on Pure and Applied Mathematics*, vol. 67, no. 6, pp. 906–956.

[23] D. L. D. S. S. Chen and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Rev.*, vol. 43, no. 3, pp. 129 –159, Mar. 2001.

[24] E. J. Candès, Y. C. Eldar, D. Needell, and P. Randall, "Compressed sensing with coherent and redundant dictionaries," *Appl. Comput. Harmon. Anal.*, vol. 31, pp. 59–73, Jul 2011.

[25] S. Nam, M. E. Davies, M. Elad, and R. Gribonval, "The cosparse analysis model and algorithms," *Applied and Computational Harmonic Analysis*, vol. 34, no. 1, pp. 30 – 56, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1063520312000450

[26] R. Rubinstein, T. Peleg, and M. Elad, "Analysis k-svd: A dictionary-learning algorithm for the analysis sparse model," *IEEE Trans. Signal Process.*, vol. 61, no. 3, pp. 661–677, Feb 2013.

[27] T. Peleg and M. Elad, "Performance guarantees of the thresholding algorithm for the cosparse analysis model," *IEEE Trans. Ind. Informat.*, vol. 59, no. 3, pp. 1832–1845, 2013.

[28] J. S. Turek, I. Yavneh, and M. Elad, "On map and mmse estimators for the co-sparse analysis model," *Digital Signal Processing*, vol. 28, pp. 57–74, May 2014.

[29] S. Nam, M. E. Davies, M. Elad, and R. Gribonval, "Cosparse analysis modeling - uniqueness and algorithms," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, 2011, pp. 5804–5807.

[30] C. Ekanadham, D. Tranchina, and E. P. Simoncelli, "Recovery of sparse translation-invariant signals with continuous basis pursuit," *IEEE Trans. Signal Process.*, vol. 59, no. 10, pp. 4735–4744, Oct 2011.

[31] T. Huang, Y. Liu, H. Meng, and X. Wang, "Adaptive matching pursuit with constrained total least squares," *EURASIP Journal on Advances in Signal Processing*, no. 1, pp. 1–12, April 2012.

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

[33] Y. C. Eldar and M. Mishali, "Robust recovery of signals from a structured union of subspaces," *IEEE Trans. Inf. Theory*, vol. 55, no. 11, pp. 5302–5316, 2009.

[34] Y. C. Eldar, P. Kuppinger, and H. Bolcskei, "Block-sparse signals: Uncertainty relations and efficient recovery," *IEEE Trans. Signal Process.*, vol. 58, no. 6, pp. 3042–3054, 2010.

[35] Y. Nesterov, "Gradient methods for minimizing composite objective function," *Mathematical Programming*, vol. 140, pp. 125–161, Aug. 2013.

[36] Z. Tan, P. Yang, and A. Nehorai, "Joint-sparse recovery in compressed sensing with dictionary mismatch," in *Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), 2013 IEEE 5th International Workshop on*, Dec 2013, pp. 248–251.

[37] P. Pal and P. P. Vaidyanathan, "Correlation-aware techniques for sparse support recovery," in *Statistical Signal Processing Workshop (SSP), 2012 IEEE*, 2012, pp. 53–56.

[38] ——, "On application of lasso for sparse support recovery with imperfect correlation awareness," in *Signals, Systems and Computers (ASILOMAR), 2012 Conference Record of the Forty Sixth Asilomar Conference on*, 2012, pp. 958–962.

[39] Z. Tan, Y. C. Eldar, A. Beck, and A. Nehorai, "Smoothing and decomposition for analysis sparse recovery," *IEEE Trans. Signal Process.*, vol. 62, no. 7, pp. 1762–1774, April 2014.

[40] E. J. Candès, "The restricted isometry property and its implications for compressed sensing," *C. R. Acad. Sci. Paris, Ser. I*, 2008.

[41] P. Pal and P. P. Vaidyanathan, "Nested arrays: A novel approach to array processing with enhanced degrees of freedom," *IEEE Trans. Signal Process.*, vol. 58, no. 8, pp. 4167–4181, 2010.

[42] P. P. Vaidyanathan and P. Pal, "Sparse sensing with co-prime samplers and arrays," *IEEE Trans. Signal Process.*, vol. 59, no. 2, pp. 573–586, 2011.

[43] A. Beck and M. Teboulle, "Smoothing and first order methods: a unified framework," *SIAM J. Optim*, vol. 22, no. 2, pp. 557–580, 2012.

[44] M. A. T. Figueiredo, R. D. Nowak, and S. J. Wright, "Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 1, no. 4, pp. 586–597, 2007.

[45] J. J. Moreau, "Proximitéet dualité dans un espace hilbertien," *Bull. Soc. Math. France*, vol. 93, pp. 273–299, 1965.

[46] H. L. V. Trees, *Optimum Array Processing: Part IV of Detection, Estimatio and Modulation Theory*. New York: Wiley Intersci., 2002.

[47] R. O. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Trans. Antennas Propag.*, vol. 34, no. 3, pp. 276–280, 1986.

[48] P. Pal and P. P. Vaidyanathan, "Coprime sampling and the music algorithm," in *Digital Signal Processing Workshop and IEEE Signal Processing Education Workshop (DSP/SPE), 2011 IEEE*, 2011, pp. 289–294.

[49] Y. D. Zhang, M. G. Amin, and B. Himed, "Sparsity-based DOA estimation using co-prime arrays," in *IEEE International Conference on Acoustics, Speech, and Signal Processing, Vancouver, Canada,*, May 2013, pp. 3967–3971.

[50] P. Pal and P. P. Vaidyanathan, "Correlation-aware sparse support recovery: Gaussian sources," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, 2013, pp. 5880–5884.

[51] Z. Tan, P. Yang, and A. Nehorai, "Joint-sparse recovery in compressed sensing with dictionary mismatch," in *The Fifth IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), IEEE*, 2013.

[52] Z. Tan and A. Nehorai, "Sparse direction of arrival estimation using co-prime arrays with off-grid targets," *IEEE Signal Process. Lett.*, vol. 21, no. 1, pp. 26–29, 2014.

[53] E. J. Candès and C. Fernandez-Granda, "Super-resolution from noisy data," *Journal of Fourier Analysis and Applications*, vol. 19, no. 6, pp. 1229–1254, Aug. 2013.

[54] C. Fernandez-Granda, "Support detection in super-resolution," in *Proceedings of the 10th International Conference on Sampling Theory and Applications*, 2013, pp. 145–148.

[55] J. A. Tropp, "Just relax: convex programming methods for identifying sparse signals in noise," *IEEE Trans. Inf. Theory*, vol. 52, no. 3, pp. 1030–1051, 2006.

[56] H. Akaike, "A new look at the statistical model identification," *IEEE Trans. Autom. Control*, vol. 19, no. 6, pp. 716–723, 1974.

[57] Z. He, A. Cichocki, S. Xie, and K. Choi, "Detecting the number of clusters in n-way probabilistic clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 11, pp. 2006–2021, 2010.

[58] W. Chen, K. M. Wong, and J. P. Reilly, "Detection of the number of signals: a predicted eigen-threshold approach," *IEEE Trans. Signal Process.*, vol. 39, no. 5, pp. 1088–1098, 1991.

[59] K. Han and A. Nehorai, "Improved source number detection and direction estimation with nested arrays and ulas using jackknifing," *IEEE Trans. Signal Process.*, vol. 61, no. 23, pp. 6118–6128, 2013.

[60] R. Prony, "Essai expérimental et analytique: sur les lois de la dilatabilité de fluides élastique et sur celles de la force expansive de la vapeur de l'alkool, à diérentes températures," *Journal de l'Ecole Polytechnique*, vol. 1, no. 2, pp. 24–76, 1795.

[61] Y. Hua and T. K. Sarkar, "Matrix pencil method for estimating parameters of exponentially damped/undamped sinusoids in noise," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 38, no. 5, pp. 814–824, May 1990.

[62] P. Bickel, Y. Ritov, and A. Tsybakov, "Simultaneous analysis of lasso and dantzig selector," *Ann. Statist.*, vol. 37, pp. 1705–1732, 2009.

[63] T. Blumensath and M. E. Davies, "Normalized iterative hard thresholding: Guaranteed stability and performance," *IEEE J. Sel. Topics Signal Process.*, vol. 4, no. 2, pp. 298–309, 2010.

[64] R. Giryes, S. Nam, M. Elad, R. Gribonval, and M. E. Davies, "Greedy-like algorithms for the cosparse analysis model," *Linear Algebra and its Applications*, vol. 441, pp. 22–60, Jan. 2014.

[65] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via alternating direction method of multipliers," in *Found. Trends Mach Learning*, vol. 3, 2010, pp. 1–122.

[66] M. V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo, "Fast image recovery using variable splitting and constrained optimization," *IEEE Trans. Image Process.*, vol. 19, no. 9, pp. 2345–2356, 2010.

[67] I. Loris and C. Verhoeven, "On a generalization of the iterative soft-thresholding algorithm for the case of non-separable penalty," *Inverse Problems*, vol. 27, no. 12, 2011.

[68] Y. E. Nesterov, "Smooth minimization of non-smooth functions," *Math. Program*, vol. 103, no. 1, pp. 127–152, 2005.

[69] M. Lustig, D. Donoho, and J. M. Pauly, "Sparse MRI: The application of compressed sensing for rapid MR imaging," *Magnetic Resonance in Medicine*, vol. 58, no. 6, pp. 1182–1195, 2007.

[70] A. Beck and M. Teboulle, "Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems," *IEEE Trans. Image Process.*, vol. 18, no. 11, pp. 2419–2434, Nov.

[71] R. Courant, "Variational methods for the solution of problems with equilibrium and vibration," *Bull. Amer. Math. Soc.*, vol. 49, pp. 1–23, 1943.

[72] Y. Wang, J. Yang, W. Yin, and Y. Zhang, "A new alternating minimization algorithm for total variation image reconstruction," *SIAM J. Imaging Sciences*, vol. 1, no. 3, pp. 248–272, 2008.

[73] J. Lin and S. Li, "Sparse recovery with coherent tight frame via analysis dantzig selector and analysis lasso," *Applied and Computational Harmonic Analysis*, Oct. 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1063520313000948

[74] J. J. Moreau, "Fonctions convexes duales et points proximaux dans un espace hilbertien." *Comptes Rendus de l'Académie des Sciences (Paris), Série A*, vol. 255, pp. 2897–2899, 1962.

[75] H. H. Bauschke and P. L. Combettes, *Convex Analysis and Monotone Operator Theory in Hilbert Space*. Springer, 2011.

[76] A. Beck and M. Teboulle, "Gradient-based algorithms with applications to signal recovery problems," in *Convex Optimization in Signal Processing and Communications*, D. Palomar and Y. Eldar, Eds. Cambridge University Press, 2009, pp. 139–162.

[77] Y. E. Nesterov, "A method for solving the convex programming problem with convergence rate $O(1/k^2)$," *Dokl. Akad. Nauk SSSR*, vol. 269, no. 3, pp. 543–547, 1983.

[78] P. J. Huber, "Robust estimation of a location parameter," *Ann. Math. Statist.*, vol. 35, pp. 73–101, 1964.

[79] D. Needell and J. A. Tropp, "Cosamp: Iterative signal recovery from noisy samples," *Appl. Comput. Harmon. Anal*, vol. 26, no. 3, pp. 301–321, 2008.

[80] N. Parikh and S. Boyd, "Proximal algorithmis," 2013.

[81] J. J. Moreau, "Proximité et dualité dans un espace hibertien," *Bull. Soc. Math. France*, vol. 93, pp. 273–299, 1965.

[82] S. Ma, W. Yin, Y. Zhang, and A. Chakraborty, "An efficient algorithm for compressed mr imaging using total variation and wavelets," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, June, pp. 1–8.

[83] E. J. Candes and T. Tao, "The dantzig selector: Statistical estimation when p is much smaller than n," *Ann. Statist.*, vol. 35, pp. 2313–2351, 2007.

[84] J. Haupt, W. U. Bajwa, G. Raz, and R. Nowak, "Toeplitz compressed sensing matrices with applications to sparse channel estimation," *IEEE Trans. Inf. Theory*, vol. 56, no. 11, pp. 5862–5875, 2010.

[85] J. S. Jiang and M. A. Ingram, "Robust detection of number of sources using the transformed rotational matrix," in *Wireless Communications and Networking Conference, 2004. WCNC. 2004 IEEE*, vol. 1, 2004, pp. 501–506 Vol.1.

[86] M. F. Duarte and R. G. Baraniuk, "Spectral compressive sensing," *Applied and Computational Harmonic Analysis*, vol. 35, no. 1, pp. 111 – 129, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1063520312001315

[87] G. Tang, B. Bhaskar, P. Shah, and B. Recht, "Compressed sensing off the grid," *IEEE Trans. Inf. Theory*, vol. 59, no. 11, pp. 7465–7490, Nov 2013.

[88] M. A. Herman and T. Strohmer, "General deviants: An analysis of perturbations in compressed sensing," *IEEE J. Sel. Topics Signal Process.*, vol. 4, no. 2, pp. 342–349, 2010.

[89] W. Chen, K. M. Wong, and J. P. Reilly, "Detection of the number of signals: a predicted eigen-threshold approach," *IEEE Trans. Signal Process.*, vol. 39, no. 5, pp. 1088–1098, 1991.

[90] R. Jagannath and K. V. S. Hari, "Block sparse estimator for grid matching in single snapshot DoA estimation," *IEEE Signal Process. Lett.*, vol. 20, no. 11, pp. 1038–1041, 2013.

[91] A. P. Petropulu, Y. Yu, and H. V. Poor, "Distributed MIMO radar using compressive sampling," in *Signals, Systems and Computers, 2008 42nd Asilomar Conference on*, Oct. 2008, pp. 203 –207.

[92] E. Fishler, A. Haimovich, R. Blum, D. Chizhik, L. Cimini, and R. Valenzuela, "MIMO radar: an idea whose time has come," in *Radar Conference, 2004. Proceedings of the IEEE*, Apr. 2004, pp. 71 – 78.

[93] J. Li, P. Stoica, L. Xu, and W. Roberts, "On parameter identifiability of MIMO radar," *IEEE Signal Process. Lett.*, vol. 14, no. 12, pp. 968 –971, Dec. 2007.

[94] T. T. Cai, L. Wang, and G. Xu, "Shifting inequality and recovery of sparse signals," *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1300–1308, 2010.

[95] D. P. Bertsekas, *Nonlinear Programming*. Athena Scientific, 2008.

[96] M. Rossi, A. Haimovich, and Y. C. Eldar, "Spatial compressive sensing for mimo radar," *arXiv:1304.4578*.

# Appendix A

# Proof of Performance Bound of Joint Sparse Recovery

Before beginning the proof of the main theorem, we give several useful lemmas which will be used in the main proof. The first lemma is based on the J-RIP property of the matrix $\boldsymbol{\Phi}$.

**Lemma A.1.** *If the matrix $\boldsymbol{\Phi}$ satisfies J-RIP with parameter $\sigma_{2K}$, then for all $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^{2N}$, which are both $K$ joint-sparse with non-overlapping support sets, we have*

$$\langle \boldsymbol{\Phi u}, \boldsymbol{\Phi v} \rangle \geq -\sigma_{2K} \|\boldsymbol{u}\|_2 \|\boldsymbol{v}\|_2.$$

**Proof:** We first consider the case when $\|\boldsymbol{u}\|_2 = 1$ and $\|\boldsymbol{v}\|_2 = 1$. According to the definition of J-RIP, we have the following inequality:

$$\begin{aligned}
\langle \boldsymbol{\Phi u}, \boldsymbol{\Phi v} \rangle =& \frac{1}{4}\{\|\boldsymbol{\Phi u} + \boldsymbol{\Phi v}\|_2^2 - \|\boldsymbol{\Phi u} - \boldsymbol{\Phi v}\|_2^2\} \\
\geq& \frac{1}{4}\{(1-\sigma_{2K})\|\boldsymbol{u}+\boldsymbol{v}\|_2^2 - (1+\sigma_{2K})\|\boldsymbol{u}-\boldsymbol{v}\|_2^2\} \\
\geq& -\sigma_{2K} + \boldsymbol{u}^{\mathrm{T}}\boldsymbol{v} \\
=& -\sigma_{2K}.
\end{aligned}$$

The last equality utilizes the fact that $\boldsymbol{u}$ and $\boldsymbol{v}$ has non-overlapping support sets. Now it is easy to extend this equation to get the result in lemma A.1.

We use $\hat{\boldsymbol{x}}$ to represent the optimal solution of (JS) and denote $\boldsymbol{x}$ as the original signal with $\boldsymbol{y} = \boldsymbol{\Phi x} + \boldsymbol{w}$; we also use $\boldsymbol{h}$ to represent the reconstruction error $\hat{\boldsymbol{x}} - \boldsymbol{x}$. Now let $\mathcal{T}$ denote

the index of coefficients with $k$ largest joint-magnitudes of vector $\boldsymbol{x}$, i.e., the indices $i$ and $N+i$ for $(1 \leq i \leq N)$ with $k$ largest $\sqrt{x_i^2 + x_{N+i}^2}$. $\mathcal{T}^c$ denotes the complement of $\mathcal{T}$. Let $\boldsymbol{x}_{\mathcal{T}}$ be a vector that maintains the same coefficients as $\boldsymbol{x}$ with support set $\mathcal{T}$, while setting other indices as zeros. Let $\mathcal{T}_0 = \mathcal{T}$, and we decompose $\mathcal{T}_0^c$ into sets of size $K$. Let $\mathcal{T}_1$ denote the locations of the $K$ largest joint-magnitudes in $\boldsymbol{h}_{\mathcal{T}^c}$, $\mathcal{T}_2$ denote the next $K$ largest joint-magnitudes in $\boldsymbol{h}_{\mathcal{T}}^c$ and so on. We also have $\mathcal{T}_{01} = \mathcal{T}_0 \cup \mathcal{T}_1$. The next lemma relates the $\ell_2$ norm of the tail to the $\ell_2/\ell_1$ norm of the tail.

**Lemma A.2.** ***(Bounding the tail)*** *For the reconstruction error $\boldsymbol{h}$ from solving (JS) and disjointed sets $\mathcal{T}_0, \mathcal{T}_1, \ldots$ defined earlier, we have*

$$\sum_{j \geq 2} \|\boldsymbol{h}_{\mathcal{T}_j}\|_2 \leq K^{-\frac{1}{2}} \|\boldsymbol{h}_{\mathcal{T}^c}\|_{2,1}.$$

**Proof :** First we can write the following inequality as

$$\|\boldsymbol{h}_{\mathcal{T}_j}\|_2 \leq K^{\frac{1}{2}} \|\boldsymbol{h}_{\mathcal{T}_j}\|_{\infty,1} \leq K^{-\frac{1}{2}} \|\boldsymbol{h}_{\mathcal{T}_{j-1}}\|_{2,1}.$$

The above equation utilizes the definition of $\|\boldsymbol{x}\|_{\infty,1}$ and also the fact that every joint magnitude in set $\mathcal{T}_j$ is no larger than every joint magnitude in set $\mathcal{T}_{j-1}$. By summing up over $j$, we obtain

$$\sum_{j \geq 2} \|\boldsymbol{h}_{\mathcal{T}_j}\|_2 \leq K^{-\frac{1}{2}} \sum_{j \geq 1} \|\boldsymbol{h}_{\mathcal{T}_j}\|_{2,1} = K^{-\frac{1}{2}} \|\boldsymbol{h}_{\mathcal{T}^c}\|_{2,1}. \tag{A.1}$$

The lemmas below are derived from the optimality of $\hat{\boldsymbol{x}}$, and they show that the reconstruction error $\boldsymbol{h}$ and $\|\boldsymbol{h}_{\mathcal{T}^c}\|_{2,1}$ is bounded when $\hat{\boldsymbol{x}}$ solves the (JS).

**Lemma A.3.** ***(Optimality condition 1)*** *Assuming that $\lambda$ obeys $\|\boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{w}\|_{\infty,1} \leq \frac{\lambda}{2}$, the reconstruction error $\boldsymbol{h}$ of (JS) satisfies the following inequality*

$$\|\boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{\Phi} \boldsymbol{h}\|_{\infty,1} \leq \frac{3}{2} \lambda,$$

**Proof**: The optimality condition for (JS) requires that the gradients vanish to zero, and it can be stated as

$$\boldsymbol{\Phi}^{\mathrm{T}} (\boldsymbol{\Phi} \hat{\boldsymbol{x}} - \boldsymbol{y}) + \lambda \boldsymbol{v} = 0, \tag{A.2}$$

where $\boldsymbol{v}$ is the gradient of function $\|\boldsymbol{v}\|_{2,1}$. It is easy to verify that $\|\boldsymbol{v}\|_{\infty,1} \leq 1$, so we get

$$\|\boldsymbol{\Phi}^{\mathrm{T}}(\boldsymbol{\Phi}\hat{\boldsymbol{x}} - \boldsymbol{y})\|_{\infty,1} = \lambda\|\boldsymbol{v}\|_{\infty,1} \leq \lambda. \tag{A.3}$$

From the assumption $\|\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{w}\|_{\infty,1} \leq \frac{\lambda}{2}$,

$$\|\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}\boldsymbol{h}\|_{\infty,1} \leq \|\boldsymbol{\Phi}^{\mathrm{T}}(\boldsymbol{\Phi}\boldsymbol{x} - \boldsymbol{y})\|_{\infty,1} + \|\boldsymbol{\Phi}^{\mathrm{T}}(\boldsymbol{\Phi}\hat{\boldsymbol{x}} - \boldsymbol{y})\|_{\infty,1} \leq \frac{3}{2}\lambda.$$

**Lemma A.4.** ***(Optimality condition 2)*** *For the reconstruction of (JS), we have following inequality:*

$$\|\boldsymbol{h}_{\mathcal{T}^c}\|_{2,1} \leq 3\|\boldsymbol{h}_{\mathcal{T}}\|_{2,1} + 4\|\boldsymbol{x}_{\mathcal{T}^c}\|_{2,1}.$$

**Proof**: Now, since $\hat{\boldsymbol{x}}$ solves the optimization problem (JS), we have

$$\frac{1}{2}\|\boldsymbol{\Phi}\hat{\boldsymbol{x}} - \boldsymbol{y}\|_2^2 + \lambda\|\hat{\boldsymbol{x}}\|_{2,1} \leq \frac{1}{2}\|\boldsymbol{\Phi}\boldsymbol{x} - \boldsymbol{y}\|_2^2 + \lambda\|\boldsymbol{x}\|_{2,1}.$$

Since $\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{x} + \boldsymbol{w}$, and by letting $\boldsymbol{h}$ denote $\hat{\boldsymbol{x}} - \boldsymbol{x}$, we have

$$\frac{1}{2}\|\boldsymbol{\Phi}\boldsymbol{h} - \boldsymbol{w}\|_2^2 + \lambda\|\hat{\boldsymbol{x}}\|_{2,1} \leq \frac{1}{2}\|\boldsymbol{w}\|_2^2 + \lambda\|\boldsymbol{x}\|_{2,1}.$$

Expanding the first term on the left side and rearranging the terms in the above equation, we get

$$\begin{aligned}\frac{1}{2}\|\boldsymbol{\Phi}\boldsymbol{h}\|_2^2 + \lambda\|\hat{\boldsymbol{x}}\|_{2,1} \leq& \langle\boldsymbol{\Phi}\boldsymbol{h}, \boldsymbol{w}\rangle + \lambda\|\boldsymbol{x}\|_{2,1} \\ \leq& \|\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{w}\|_{\infty,1}\|\boldsymbol{h}\|_{2,1} + \lambda\|\boldsymbol{x}\|_{2,1}.\end{aligned}$$

The second inequality follows from the fact that

$$\begin{aligned}\langle\boldsymbol{x}, \boldsymbol{y}\rangle =& \sum_{i=1}^{N}(x_i y_i + x_{N+i}y_{N+i}) \\ \leq& \sum_{i=1}^{N}\sqrt{x_i^2 + x_{N+i}^2}\sqrt{y_i^2 + y_{N+i}^2} \\ \leq& \|\boldsymbol{x}\|_{2,1}\|\boldsymbol{y}\|_{\infty,1}.\end{aligned}$$

With the assumption that $\|\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{w}\|_{\infty,1} \leq \frac{\lambda}{2}$, we get

$$\frac{1}{2}\|\boldsymbol{\Phi}\boldsymbol{h}\|_2^2 + \lambda\|\hat{\boldsymbol{x}}\|_{2,1} \leq \frac{\lambda}{2}\|\boldsymbol{h}\|_{2,1} + \lambda\|\boldsymbol{x}\|_{2,1}.$$

Therefore we have

$$\lambda\|\hat{\boldsymbol{x}}\|_{2,1} \leq \frac{1}{2}\|\boldsymbol{\Phi}\boldsymbol{h}\|_2^2 + \lambda\|\hat{\boldsymbol{x}}\|_{2,1} \leq \frac{\lambda}{2}\|\boldsymbol{h}\|_{2,1} + \lambda\|\boldsymbol{x}\|_{2,1}.$$

Since we have $\boldsymbol{h} = \hat{\boldsymbol{x}} - \boldsymbol{x}$, we also have

$$\|\boldsymbol{h} + \boldsymbol{x}\|_{2,1} \leq \frac{1}{2}\|\boldsymbol{h}\|_{2,1} + \|\boldsymbol{x}\|_{2,1}.$$

Using the above equation, we can show that

$$\|\boldsymbol{h}_{\mathcal{T}} + \boldsymbol{x}_{\mathcal{T}}\|_{2,1} + \|\boldsymbol{h}_{\mathcal{T}^c} + \boldsymbol{x}_{\mathcal{T}^c}\|_{2,1} \leq \frac{1}{2}\|\boldsymbol{h}_{\mathcal{T}}\|_{2,1} + \frac{1}{2}\|\boldsymbol{h}_{\mathcal{T}^c}\|_{2,1} + \|\boldsymbol{x}_{\mathcal{T}}\|_{2,1} + \|\boldsymbol{x}_{\mathcal{T}^c}\|_{2,1}.$$

Applying triangle inequality on the left hand side of above inequality, we have

$$-\|\boldsymbol{h}_{\mathcal{T}}\|_{2,1} + \|\boldsymbol{x}_{\mathcal{T}}\|_{2,1} + \|\boldsymbol{h}_{\mathcal{T}^c}\|_{2,1} - \|\boldsymbol{x}_{\mathcal{T}^c}\|_{2,1} \leq \frac{1}{2}\|\boldsymbol{h}_{\mathcal{T}}\|_{2,1} + \frac{1}{2}\|\boldsymbol{h}_{\mathcal{T}^c}\|_{2,1} + \|\boldsymbol{x}_{\mathcal{T}}\|_{2,1} + \|\boldsymbol{x}_{\mathcal{T}^c}\|_{2,1}.$$

After rearranging the terms, we have the following cone constraint:

$$\|\boldsymbol{h}_{\mathcal{T}^c}\|_{2,1} \leq 3\|\boldsymbol{h}_{\mathcal{T}}\|_{2,1} + 4\|\boldsymbol{x}_{\mathcal{T}^c}\|_{2,1}. \tag{A.4}$$

With the above lemmas, we can prove theorem 2.2 as follows.

**Main Proof:** The proof follows some techniques in [33], [51] and [40]. The challenge lies in two aspects. First, instead of dealing with sparsity, we have to use the property of joint-sparsity for the derivation. Second, unlike the constrained optimization considered in [33], in this work we are trying to derive the performance bound for an unconstrained optimization. The proof is derived in two steps. First, we show that $\boldsymbol{h}$ inside the set $\mathcal{T}_{01}$ is bounded by the

terms of $h$ outside the set $\mathcal{T}$. Then we show that $\boldsymbol{h}_{\mathcal{T}^c}$ is essentially small. First we have

$$\begin{aligned}
\langle \boldsymbol{\Phi h}, \boldsymbol{\Phi h}_{\mathcal{T}_{01}} \rangle =& \langle \boldsymbol{\Phi h}_{\mathcal{T}_{01}}, \boldsymbol{\Phi h}_{\mathcal{T}_{01}} \rangle + \sum_{j\geq 2} \langle \boldsymbol{\Phi h}_{\mathcal{T}_j}, \boldsymbol{\Phi h}_{\mathcal{T}_{01}} \rangle \\
\geq& (1-\sigma_{2K}) \|\boldsymbol{h}_{\mathcal{T}_{01}}\|_2^2 + \sum_{j\geq 2} \langle \boldsymbol{\Phi h}_{\mathcal{T}_j}, \boldsymbol{\Phi h}_{\mathcal{T}_0} \rangle + \sum_{j\geq 2} \langle \boldsymbol{\Phi h}_{\mathcal{T}_j}, \boldsymbol{\Phi h}_{\mathcal{T}_1} \rangle \\
\geq& (1-\sigma_{2K}) \|\boldsymbol{h}_{\mathcal{T}_{01}}\|_2^2 - \sigma_{2K} \|\boldsymbol{h}_{\mathcal{T}_0}\|_2 \sum_{j\geq 2} \|\boldsymbol{h}_{\mathcal{T}_j}\|_2 - \sigma_{2K} \|\boldsymbol{h}_{\mathcal{T}_1}\|_2 \sum_{j\geq 2} \|\boldsymbol{h}_{\mathcal{T}_j}\|_2 \\
=& (1-\sigma_{2K}) \|\boldsymbol{h}_{\mathcal{T}_{01}}\|_2^2 - \sigma_{2K} (\|\boldsymbol{h}_{\mathcal{T}_0}\|_2 + \|\boldsymbol{h}_{\mathcal{T}_1}\|_2) \sum_{j\geq 2} \|\boldsymbol{h}_{\mathcal{T}_j}\|_2 \\
\geq& (1-\sigma_{2K}) \|\boldsymbol{h}_{\mathcal{T}_{01}}\|_2^2 - \sqrt{2}\sigma_{2K} \|\boldsymbol{h}_{\mathcal{T}_{01}}\|_2 \sum_{j\geq 2} \|\boldsymbol{h}_{\mathcal{T}_j}\|_2.
\end{aligned}$$

The first inequality follows the J-RIP of matrix $\boldsymbol{\Phi}$. The second inequality uses the result from Lemma A.1. The third one is deduced from the fact that $\|\boldsymbol{h}_{\mathcal{T}_0}\|_2 + \|\boldsymbol{h}_{\mathcal{T}_1}\|_2 \leq \sqrt{2}\|\boldsymbol{h}_{\mathcal{T}_{01}}\|_2$ when the set $\mathcal{T}_0$ and the set $\mathcal{T}_1$ are disjoint. With the result from Lemma A.2, we have our final inequality as

$$\langle \boldsymbol{\Phi h}, \boldsymbol{\Phi h}_{\mathcal{T}_{01}} \rangle \geq (1-\sigma_{2K}) \|\boldsymbol{h}_{\mathcal{T}_{01}}\|_2^2 - \sqrt{2} K^{-\frac{1}{2}} \sigma_{2K} \|\boldsymbol{h}_{\mathcal{T}_{01}}\|_2 \|\boldsymbol{h}_{\mathcal{T}^c}\|_{2,1}. \tag{A.5}$$

From the inequality $\langle \boldsymbol{x}, \boldsymbol{y} \rangle = \sum_{i=1}^N (x_i y_i + x_{N+i} y_{N+i}) \leq \sum_{i=1}^N \sqrt{x_i^2 + x_{N+i}^2} \sqrt{y_i^2 + y_{N+i}^2} \leq \|\boldsymbol{x}\|_{2,1} \|\boldsymbol{y}\|_{\infty,1}$, we get

$$\begin{aligned}
\langle \boldsymbol{\Phi h}, \boldsymbol{\Phi h}_{\mathcal{T}_{01}} \rangle =& \langle \boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{\Phi h}, \boldsymbol{h}_{\mathcal{T}_{01}} \rangle \leq \|\boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{\Phi h}\|_{\infty,1} \|\boldsymbol{h}_{\mathcal{T}_{01}}\|_{2,1} \\
\leq& \sqrt{2K} \|\boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{\Phi h}\|_{\infty,1} \|\boldsymbol{h}_{\mathcal{T}_{01}}\|_2 \leq \sqrt{K} c_0 \lambda \|\boldsymbol{h}_{\mathcal{T}_{01}}\|_2,
\end{aligned} \tag{A.6}$$

where $c_0 = \frac{3\sqrt{2}}{2}$. The second inequality uses the fact that $\|\boldsymbol{h}_{\mathcal{T}_{01}}\|_{2,1} \leq \sqrt{2K} \|\boldsymbol{h}_{\mathcal{T}_{01}}\|_2$, which is derived by using Cauchy-Schwarz inequality. The last inequality follows the result of Lemma A.3. Combining equations (A.5) and (A.6), we get

$$\|\boldsymbol{h}_{\mathcal{T}_{01}}\|_2 \leq \frac{\sqrt{K}\lambda c_0 + \sqrt{2} K^{-\frac{1}{2}} \sigma_{2K} \|\boldsymbol{h}_{\mathcal{T}^c}\|_{2,1}}{1-\sigma_{2K}}. \tag{A.7}$$

Hence, combining the Cauchy-Schwarz inequality with the result from last inequality leads to

$$\|\boldsymbol{h}_{\mathcal{T}}\|_{2,1} \leq \sqrt{K}\|\boldsymbol{h}_{\mathcal{T}}\|_2 \leq \sqrt{K}\|\boldsymbol{h}_{\mathcal{T}_{01}}\|_2 \leq \frac{\lambda K c_0 + \sqrt{2}\sigma_{2K}\|\boldsymbol{h}_{\mathcal{T}^c}\|_{2,1}}{1-\sigma_{2K}}. \tag{A.8}$$

Next, we prove that $\boldsymbol{h}_{\mathcal{T}^c}$ is relatively small. Combining the inequalities from Lemma A.4 and (A.8), we have

$$\|\boldsymbol{h}_{\mathcal{T}^c}\|_{2,1} \leq \frac{3\lambda K c_0 + 3\sqrt{2}\sigma_{2K}\|\boldsymbol{h}_{\mathcal{T}^c}\|_{2,1}}{1-\sigma_{2K}} + 4\|\boldsymbol{x}_{\mathcal{T}^c}\|_{2,1}.$$

From the assumption $\sigma_{2K} < 0.1907$, we have $1-(1+3\sqrt{2})\sigma_{2K} > 0$. Then by rearranging the terms, the above inequality becomes

$$\|\boldsymbol{h}_{\mathcal{T}^c}\|_{2,1} \leq \frac{3\lambda K c_0 + 4(1-\sigma_{2K})\|\boldsymbol{x}_{\mathcal{T}^c}\|_{2,1}}{1-(1+3\sqrt{2})\sigma_{2K}}. \tag{A.9}$$

Now we can bound the reconstruction error $\boldsymbol{h}$. Using the results from Lemma A.2 and equations (A.7) and (A.9), we derive

$$\begin{aligned}
\|\boldsymbol{h}\|_2 \leq & \|\boldsymbol{h}_{\mathcal{T}_{01}}\|_2 + \sum_{j\geq 2}\|\boldsymbol{h}_{\mathcal{T}_j}\|_2 \\
\leq & \frac{\sqrt{K}\lambda c_0 + \sqrt{2}K^{-\frac{1}{2}}\sigma_{2K}\|\boldsymbol{h}_{\mathcal{T}^c}\|_{2,1}}{1-\sigma_{2K}} + K^{-\frac{1}{2}}\|\boldsymbol{h}_{\mathcal{T}^c}\|_{2,1} \\
= & \frac{c_0\lambda\sqrt{K}}{1-\sigma_{2K}} + \frac{((\sqrt{2}-1)\sigma_{2K}+1)K^{-\frac{1}{2}}\|\boldsymbol{h}_{\mathcal{T}^c}\|_{2,1}}{1-\sigma_{2K}} \\
\leq & C_0\sqrt{K}\lambda + C_1\frac{\|\boldsymbol{x}-(\boldsymbol{x})_K\|_{2,1}}{\sqrt{K}}.
\end{aligned} \tag{A.10}$$

The first inequality uses the triangle inequality. For the second inequality we use Lemma A.2. The constants are given as

$$C_0 = \frac{6\sqrt{2}}{1-(1+3\sqrt{2})\sigma_{2K}}, \quad C_1 = \frac{4((\sqrt{2}-1)\sigma_{2K}+1)}{1-(1+3\sqrt{2})\sigma_{2K}}.$$

# Appendix B

# Statistical Analysis of the Co-prime Arrays

To derive the statistical behavior of each element in $\boldsymbol{E}$ in Lemma 3.1 we rely on two lemmas regarding the concentration behavior of complex Gaussian random variables. Their proofs are based on results from [84].

**Lemma B.1.** *Let $x(t)$ and $y(t), t = 1, \ldots, T$ be sequences of i.i.d., circularly-symmetric complex normal variables with zero mean and variances equal to $\sigma_x^2$ and $\sigma_y^2$ respectively. That is $x(t) \sim \mathcal{CN}(0, \sigma_x^2)$ and $y(t) \sim \mathcal{CN}(0, \sigma_y^2)$. Then*

$$\Pr\left(\left|\sum_{t=1}^{T} x(t)y^*(t)\right| \geq \epsilon\right) \leq 8\exp\left(-\frac{\epsilon^2}{16\sigma_x\sigma_y(T\sigma_x\sigma_y + \frac{\epsilon}{4})}\right).$$

**Proof:** First we have

$$\sum_{t=1}^{T} x(t)y^*(t) = \sum_{t=1}^{T} \mathrm{Re}[x(t)]\mathrm{Re}[y(t)] + \sum_{t=1}^{T} \mathrm{Im}[x(t)]\mathrm{Im}[y(t)]$$
$$- \boldsymbol{j}\sum_{t=1}^{T} \mathrm{Re}[x(t)]\mathrm{Im}[y(t)] + \boldsymbol{j}\sum_{t=1}^{T} \mathrm{Im}[x(t)]\mathrm{Re}[y(t)].$$

Following the same procedure used in the proof of Lemma 3.1, we have

$$\begin{aligned}\Pr\left(\left|\sum_{t=1}^{T} x(t)y^*(t)\right| \geq \epsilon\right) \leq & \Pr\left(\left|\sum_{t=1}^{T} \mathrm{Re}[x(t)]\mathrm{Re}[y(t)]\right| \geq \frac{\epsilon}{4}\right) + \Pr\left(\left|\sum_{t=1}^{T} \mathrm{Im}[x(t)]\mathrm{Im}[y(t)]\right| \geq \frac{\epsilon}{4}\right) \\ & + \Pr\left(\left|\sum_{t=1}^{T} \mathrm{Re}[x(t)]\mathrm{Im}[y(t)]\right| \geq \frac{\epsilon}{4}\right) + \Pr\left(\left|\sum_{t=1}^{T} \mathrm{Im}[x(t)]\mathrm{Re}[y(t)]\right| \geq \frac{\epsilon}{4}\right).\end{aligned}$$

Applying Lemma 6 from [84] concludes the proof. □

Before introducing the next lemma, we need to show that the square sums of i.i.d Gaussian random variables concentrate around the sum of their variances. The results below rely on Lemma 7 from [84].

**Lemma B.2.** *Let $x(t), t = 1, \ldots, T$ be a sequence of i.i.d. Gaussian random variables with zero mean and variance equal to $\sigma^2$, i.e., $x(t) \sim \mathcal{N}(0, \sigma^2)$. Then*

$$\Pr\left(\left|\sum_{t=1}^{T} x^2(t) - T\sigma^2\right| \geq \epsilon\right) \leq 2\exp\left(-\frac{\epsilon^2}{16\sigma^4 T}\right)$$

*for $0 \leq \epsilon \leq 4\sigma^2 T$.*

**Proof:** From the results in [84], for any positive $c$, we have the asymmetric bounds

$$\Pr\left(\sum_{t=1}^{T} x^2(t) - T\sigma^2 \geq 2\sigma^2\sqrt{Tc} + 2\sigma^2 c\right) \leq \exp(-c),$$

$$\Pr\left(\sum_{t=1}^{T} x^2(t) - T\sigma^2 \leq -2\sigma^2\sqrt{Tc}\right) \leq \exp(-c).$$

When $0 \leq c \leq T$, we obtain

$$\Pr\left(\sum_{t=1}^{T} x^2(t) - T\sigma^2 \geq 4\sigma^2\sqrt{Tc}\right) \leq \exp(-c),$$

$$\Pr\left(\sum_{t=1}^{T} x^2(t) - T\sigma^2 \leq -4\sigma^2\sqrt{Tc}\right) \leq \exp(-c).$$

Combing the above two inequalities leads to

$$\Pr\left(\left|\sum_{t=1}^{T} x^2(t) - T\sigma^2\right| \geq 4\sigma^2\sqrt{Tc}\right) \leq 2\exp(-c),$$

which yields the result by replacing $4\sigma^2\sqrt{Tc}$ with $\epsilon$ while maintaining $0 \leq c \leq T$. $\square$

**Lemma B.3.** *Let $x(t) \sim \mathcal{CN}(0, \sigma_x^2), t = 1, \ldots, T$ be a sequence of i.i.d., circularly-symmetric complex normal random variable. When $0 \leq \epsilon \leq 4\sigma_x^2 T$, we have*

$$\Pr\left(\left|\sum_{t=1}^{T} |x(t)|^2 - T\sigma_x^2\right| \geq \epsilon\right) \leq 4\exp\left(-\frac{\epsilon^2}{16T\sigma_x^4}\right).$$

**Proof:** We begin by noting that

$$\sum_{t=1}^{T} |x(t)|^2 - T\sigma_x^2 = \sum_{t=1}^{T} [\mathrm{Re}\, x(t)]^2 + \sum_{t=1}^{T} [\mathrm{Im}\, x(t)]^2 - T\sigma_x^2.$$

Therefore

$$\Pr\left(\left|\sum_{t=1}^{T} |x(t)|^2 - T\sigma_x^2\right| \geq \epsilon\right) \leq \Pr\left(\left|\sum_{t=1}^{T} [\mathrm{Re}\, x(t)]^2 - \frac{T\sigma_x^2}{2}\right| \geq \frac{\epsilon}{2}\right) + \Pr\left(\left|\sum_{t=1}^{T} [\mathrm{Im}\, x(t)]^2 - \frac{T\sigma_x^2}{2}\right| \geq \frac{\epsilon}{2}\right).$$

Applying Lemma B.2, we establish the result. $\square$

**Proof of Lemma 3.1:**

We use $T_1, T_2$, and $T_3$ to denote the first three terms in (3.17). The last two terms are denoted by $T_4$. Then

$$\begin{aligned}\Pr(|E_{mn}| \leq \epsilon) &\geq \Pr\left(\cap_{i=1}^{4} |T_i| \leq \frac{\epsilon}{4}\right) = 1 - \Pr\left(\cup_{i=1}^{4} |T_i| \geq \frac{\epsilon}{4}\right) \\ &\geq 1 - \sum_{i=1}^{4} \Pr\left(|T_i| \geq \frac{\epsilon}{4}\right),\end{aligned}$$

which leads to the inequality

$$\Pr(|E_{mn}| \geq \epsilon) \leq \sum_{i=1}^{4} \Pr\left(|T_i| \geq \frac{\epsilon}{4}\right). \tag{B.1}$$

We also have

$$\begin{aligned} |T_1| &= \frac{1}{T} \sum_{t=1}^{T} \sum_{i,j=1, i\neq j}^{K} A_{mi} A_{nj}^* s_i(t) s_j^*(t) \\ &\leq \frac{1}{T} \sum_{i,j=1, i\neq j}^{K} |A_{mi} A_{ni}^*| \left| \sum_{t=1}^{T} s_i(t) s_j^*(t) \right| \\ &\leq \frac{1}{T} \sum_{i,j=1, i\neq j}^{K} \left| \sum_{t=1}^{T} s_i(t) s_j^*(t) \right|. \end{aligned} \tag{B.2}$$

The last inequality follows from the fact that $|A_{mn}| \leq 1$ for all $m, n$. Thus

$$\Pr\left(|T_1| \geq \frac{\epsilon}{4}\right) \leq \Pr\left( \sum_{i,j=1, i\neq j}^{K} \left| \sum_{t=1}^{T} s_i(t) s_j^*(t) \right| \geq \frac{\epsilon T}{4} \right).$$

Clearly

$$\Pr\left(|T_1| \geq \frac{\epsilon}{4}\right) \leq \Pr\left( \left| \sum_{t=1}^{T} s_{i_0}(t) s_{j_0}^*(t) \right| \geq \frac{\epsilon T}{4K(K-1)} \right),$$

for some $i_0, j_0$ with $i_0 \neq j_0$. Using Lemma B.1

$$\Pr\left(|T_1| \geq \frac{\epsilon}{4}\right) \leq 8 \exp(-C_1(\epsilon) T), \tag{B.3}$$

with $C_1(\epsilon) = \frac{\epsilon^2}{16\sigma_s^2 K(K-1)(16\sigma_s^2 K(K-1)+\epsilon)}$.

For the second term $T_2$, we have

$$|T_2| = \frac{1}{T} \sum_{t=1}^{T} \sum_{i=1}^{K} A_{mi} s_i(t) \varepsilon_n^*(t) \leq \frac{1}{T} \sum_{i=1}^{K} |A_{mi}| \left| \sum_{t=1}^{T} s_i(t) \varepsilon_n(t)^* \right| \leq \frac{1}{T} \sum_{i=1}^{K} \left| \sum_{t=1}^{T} s_i(t) \varepsilon_n(t)^* \right|. \tag{B.4}$$

Following similar arguments as for $T_1$, we obtain that

$$\Pr\left(|T_2| \geq \frac{\epsilon}{4}\right) \leq \Pr\left(\left|\sum_{t=1}^{T} s_{i_0}(t)\varepsilon_n^*(t)\right| \geq \frac{\epsilon T}{4K}\right).$$

Applying Lemma B.1, we have

$$\Pr\left(|T_2| \geq \frac{\epsilon}{4}\right) \leq 8\exp(-C_2(\epsilon)T), \tag{B.5}$$

with $C_2(\epsilon) = \frac{\epsilon^2}{16\sigma_s\sigma K(16\sigma_s\sigma K+\epsilon)}$.

For the third term, we have the same results as the second one, given as

$$\Pr\left(|T_3| \geq \frac{\epsilon}{4}\right) \leq 8\exp(-C_2(\epsilon)T). \tag{B.6}$$

When $m \neq n$, the last term $T_4 = \frac{1}{T}\sum_{t=1}^{T} \varepsilon_m(t)\varepsilon_n^*(t)$, and by Lemma B.1,

$$\Pr\left(|T_4| \geq \frac{\epsilon}{4}\right) \leq 8\exp(-C_3(\epsilon)T), \tag{B.7}$$

with $C_3(\epsilon) = \frac{\epsilon^2}{16\sigma^2(16\sigma^2+\epsilon)}$. When $m = n$, the last term is given as $T_4 = \frac{1}{T}\sum_{t=1}^{T} |\varepsilon_m(t)|^2 - \sigma^2$, thus the probability is bounded by

$$\Pr\left(|T_4| \geq \frac{\epsilon}{4}\right) \leq 4\exp(-C_4(\epsilon)T), \tag{B.8}$$

where $C_4(\epsilon) = \frac{\epsilon^2}{256\sigma_\varepsilon^2}$ and $\epsilon \leq 16\sigma^2$ according to Lemma B.3. Applying the results from (B.3), (B.5), (B.6), (B.7) and (B.8) to inequality (B.1), we obtain the desired result. □

# Appendix C

# Derivation of the Dual Problem

By introducing the variable $\boldsymbol{z} \in \mathbb{C}^{2MN+1}$, the original primal problem is equivalent to the following optimization:

$$\begin{aligned} &\min_{\boldsymbol{s},\sigma^2\geq 0,\boldsymbol{z}} \|\boldsymbol{s}\|_{\mathrm{TV}} \\ \text{s.t.} \quad &\|\boldsymbol{z}\|_2 \leq \epsilon, \quad \boldsymbol{z} = \boldsymbol{r} - \boldsymbol{F}\boldsymbol{s} - \sigma^2\boldsymbol{w}. \end{aligned}$$

With the Lagrangian multiplier $v \geq 0$ and $\boldsymbol{u} \in \mathbb{C}^{2MN+1}$, the Lagrangian function is given as

$$L(\boldsymbol{s},\boldsymbol{z},\sigma^2,\boldsymbol{u},v) = \|\boldsymbol{s}\|_{\mathrm{TV}} + v(\|\boldsymbol{z}\|_2 - \epsilon) + \mathrm{Re}[\boldsymbol{u}^*(\boldsymbol{r} - \boldsymbol{F}\boldsymbol{s} - \sigma^2\boldsymbol{w} - \boldsymbol{z})].$$

The dual function is given as

$$g(\boldsymbol{u},v) = \mathrm{Re}[\boldsymbol{u}^*\boldsymbol{r}] - v\epsilon + \inf_{\boldsymbol{s},\boldsymbol{z},\sigma^2\geq 0}\{\|\boldsymbol{s}\|_{\mathrm{TV}} - \mathrm{Re}[\boldsymbol{u}^*\boldsymbol{F}\boldsymbol{s}] - \sigma^2\mathrm{Re}[\boldsymbol{u}^*\boldsymbol{w}] + v\|\boldsymbol{z}\|_2 - \boldsymbol{u}^*\boldsymbol{z}\}.$$

The Lagrangian multipliers $\boldsymbol{u}$ and $v$ in the domain of the dual function have to satisfy the following three constraints:

$$\|\boldsymbol{F}^*\boldsymbol{u}\|_{L_\infty} \leq 1, \mathrm{Re}[\boldsymbol{u}^*\boldsymbol{w}] \leq 0, v\frac{\boldsymbol{z}}{\|\boldsymbol{z}\|_2} = \boldsymbol{u}.$$

From the third constraint, we have $v = \|\boldsymbol{u}\|_2$, resulting in the dual problem stated in (3.33).

# Appendix D

# Proof of Lemmas for the Analysis Model

**Proof of Lemma 4.2:** Without loss of generality we assume that $\|\boldsymbol{u}\|_2 = 1$ and $\|\boldsymbol{v}\|_2 = 1$. By the definition of D-RIP, we have

$$\begin{aligned}\mathrm{Re}\langle \boldsymbol{Au}, \boldsymbol{Av}\rangle =& \frac{1}{4}\{\|\boldsymbol{Au}+\boldsymbol{Av}\|_2^2 - \|\boldsymbol{Au}-\boldsymbol{Av}\|_2^2\} \\ \geq& \frac{1}{4}\{(1-\sigma_{2s})\|\boldsymbol{u}+\boldsymbol{v}\|_2^2 - (1+\sigma_{2s})\|\boldsymbol{u}-\boldsymbol{v}\|_2^2\} \\ =& -\sigma_{2s} + \mathrm{Re}\langle \boldsymbol{u}, \boldsymbol{v}\rangle.\end{aligned}$$

Now it is easy to extend this equation to get the desired result.

**Proof of Lemma 4.3:** From the definition of $\mathcal{T}_j$ we have

$$\|\boldsymbol{D}^*_{\mathcal{T}_j}\boldsymbol{h}\|_2 \leq s^{-\frac{1}{2}}\|\boldsymbol{D}^*_{\mathcal{T}_{j-1}}\boldsymbol{h}\|_1$$

for all $j \geq 2$. Summing $j = 2, 3, \ldots$ leads to

$$\sum_{j\geq 2}\|\boldsymbol{D}^*_{\mathcal{T}_j}\boldsymbol{h}\|_2 \leq s^{-\frac{1}{2}}\sum_{j\geq 1}\|\boldsymbol{D}^*_{\mathcal{T}_j}\boldsymbol{h}\|_1 = s^{-\frac{1}{2}}\|\boldsymbol{D}^*_{\mathcal{T}^c}\boldsymbol{h}\|_1. \tag{D.1}$$

Now, considering the fact that $\boldsymbol{D}$ is a tight frame, i.e., $\boldsymbol{D}\boldsymbol{D}^* = \boldsymbol{I}$, and that the D-RIP holds,

$$\begin{aligned}
&\mathrm{Re}\langle \boldsymbol{A}\boldsymbol{h}, \boldsymbol{A}\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\rangle \\
=&\mathrm{Re}\langle \boldsymbol{A}\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}, \boldsymbol{A}\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\rangle + \sum_{j\geq 2}\mathrm{Re}\langle \boldsymbol{A}\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_j}\boldsymbol{h}, \boldsymbol{A}\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\rangle \\
\geq&(1-\sigma_{2s})\|\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\|_2^2 + \sum_{j\geq 2}\mathrm{Re}\langle \boldsymbol{A}\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_j}\boldsymbol{h}, \boldsymbol{A}\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_0}\boldsymbol{h}\rangle + \sum_{j\geq 2}\mathrm{Re}\langle \boldsymbol{A}\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_j}\boldsymbol{h}, \boldsymbol{A}\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_1}\boldsymbol{h}\rangle
\end{aligned}$$

Using the result from Lemma 4.2, we can bound the last two terms in the above inequality; hence, we derive

$$\begin{aligned}
&\mathrm{Re}\langle \boldsymbol{A}\boldsymbol{h}, \boldsymbol{A}\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\rangle \\
\geq&(1-\sigma_{2s})\|\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\|_2^2 + \sum_{j\geq 2}\mathrm{Re}\langle \boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_j}\boldsymbol{h}, \boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_0}\boldsymbol{h}\rangle + \sum_{j\geq 2}\mathrm{Re}\langle \boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_j}\boldsymbol{h}, \boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_1}\boldsymbol{h}\rangle \\
&- \sigma_{2s}\|\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_0}\boldsymbol{h}\|_2 \sum_{j\geq 2}\|\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_j}\boldsymbol{h}\|_2 - \sigma_{2s}\|\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_1}\boldsymbol{h}\|_2 \sum_{j\geq 2}\|\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_j}\boldsymbol{h}\|_2 \\
=&(1-\sigma_{2s})\|\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\|_2^2 + \mathrm{Re}\left\langle \sum_{j\geq 2}\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_j}\boldsymbol{h}, \boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\right\rangle \\
&- \sigma_{2s}(\|\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_0}\boldsymbol{h}\|_2 + \|\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_1}\boldsymbol{h}\|_2) \sum_{j\geq 2}\|\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_j}\boldsymbol{h}\|_2 \qquad \text{(D.2)}
\end{aligned}$$

By definition of $\mathcal{T}_j$, we have

$$\begin{aligned}
\mathrm{Re}\left\langle \sum_{j\geq 2}\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_j}\boldsymbol{h}, \boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\right\rangle &= \mathrm{Re}\langle \boldsymbol{h} - \boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}, \boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\rangle \\
&= \|\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\|_2^2 - \|\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\|_2^2.
\end{aligned}$$

Combining this equation with (D.2) results in

$$\begin{aligned}
&\mathrm{Re}\langle \boldsymbol{A}\boldsymbol{h}, \boldsymbol{A}\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\rangle \\
\geq&\|\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\|_2^2 - \sigma_{2s}\|\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\|_2^2 + \|\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\|_2^2 - \|\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\|_2^2 \\
&- \sigma_{2s}(\|\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_0}\boldsymbol{h}\|_2 + \|\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_1}\boldsymbol{h}\|_2) \sum_{j\geq 2}\|\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_j}\boldsymbol{h}\|_2.
\end{aligned}$$

Using the fact that when $\boldsymbol{D}$ is a tight frame, $\|\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\|_2 \leq \|\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\|_2$, we have

$$\begin{aligned}&\mathrm{Re}\langle \boldsymbol{A}\boldsymbol{h}, \boldsymbol{A}\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\rangle\\ \geq&(1-\sigma_{2s})\|\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\|_2^2 - \sigma_{2s}(\|\boldsymbol{D}^*_{\mathcal{T}_0}\boldsymbol{h}\|_2 + \|\boldsymbol{D}^*_{\mathcal{T}_1}\boldsymbol{h}\|_2)\sum_{j\geq 2}\|\boldsymbol{D}^*_{\mathcal{T}_j}\boldsymbol{h}\|_2.\end{aligned}$$

Since $\|\boldsymbol{D}^*_{\mathcal{T}_0}\boldsymbol{h}\|_2 + \|\boldsymbol{D}^*_{\mathcal{T}_1}\boldsymbol{h}\|_2 \leq \sqrt{2}\|\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\|_2$ (becuase $\mathcal{T}_0$ and $\mathcal{T}_1$ are disjoint), we conclude that

$$\begin{aligned}&\mathrm{Re}\langle \boldsymbol{A}\boldsymbol{h}, \boldsymbol{A}\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\rangle\\ \geq&(1-\sigma_{2s})\|\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\|_2^2 - \sqrt{2}\sigma_{2s}\|\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\|_2\sum_{j\geq 2}\|\boldsymbol{D}^*_{\mathcal{T}_j}\boldsymbol{h}\|_2,\end{aligned}$$

which along with inequality (D.1) yields the desired result given by

$$\begin{aligned}&\mathrm{Re}\langle \boldsymbol{A}\boldsymbol{h}, \boldsymbol{A}\boldsymbol{D}\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\rangle\\ \geq&(1-\sigma_{2s})\|\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\|_2^2 - \sqrt{2}s^{-\frac{1}{2}}\sigma_{2s}\|\boldsymbol{D}^*_{\mathcal{T}_{01}}\boldsymbol{h}\|_2\|\boldsymbol{D}^*_{\mathcal{T}^c}\boldsymbol{h}\|_1.\end{aligned}$$

**Proof of Lemma 4.4**: The subgradient optimality condition for RALASSO (4.5) can be stated as

$$\boldsymbol{A}^*(\boldsymbol{A}\hat{\boldsymbol{x}}_\rho - \boldsymbol{b}) + \rho\boldsymbol{D}(\boldsymbol{D}^*\hat{\boldsymbol{x}}_\rho - \hat{\boldsymbol{z}}_\rho) = 0, \tag{D.3}$$

$$\lambda\boldsymbol{v} + \rho(\hat{\boldsymbol{z}}_\rho - \boldsymbol{D}^*\hat{\boldsymbol{x}}_\rho) = 0, \tag{D.4}$$

where $\boldsymbol{v}$ is a subgradient of the function $\|\boldsymbol{z}\|_1$ and consequently $\|\boldsymbol{v}\|_\infty \leq 1$. Combining (D.3) and (D.4), we have

$$\boldsymbol{A}^*(\boldsymbol{A}\hat{\boldsymbol{x}}_\rho - \boldsymbol{b}) = \lambda\boldsymbol{D}\boldsymbol{v}.$$

Multiplying both sides by $\boldsymbol{D}^*$, we get

$$\|\boldsymbol{D}^*\boldsymbol{A}^*(\boldsymbol{A}\hat{\boldsymbol{x}}_\rho - \boldsymbol{b})\|_\infty = \lambda\|\boldsymbol{D}^*\boldsymbol{D}\boldsymbol{v}\|_\infty \leq \lambda\|\boldsymbol{D}^*\boldsymbol{D}\|_{\infty,\infty} = \lambda\|\boldsymbol{D}^*\boldsymbol{D}\|_{1,1}. \tag{D.5}$$

The first inequality follows from the fact that $\|\boldsymbol{v}\|_\infty \leq 1$. With the assumption that $\|\boldsymbol{D}^*\boldsymbol{A}^*\boldsymbol{w}\|_\infty \leq \frac{\lambda}{2}$, and the triangle inequality, we have

$$\begin{aligned}
&\|\boldsymbol{D}^*\boldsymbol{A}^*\boldsymbol{A}\boldsymbol{h}\|_\infty \\
&\leq \|\boldsymbol{D}^*\boldsymbol{A}^*(\boldsymbol{A}\boldsymbol{x}-\boldsymbol{b})\|_\infty + \|\boldsymbol{D}^*\boldsymbol{A}^*(\boldsymbol{A}\hat{\boldsymbol{x}}_\rho-\boldsymbol{b})\|_\infty \\
&\leq \left(\frac{1}{2} + \|\boldsymbol{D}^*\boldsymbol{D}\|_{1,1}\right)\lambda. \qquad \text{(D.6)}
\end{aligned}$$

**Proof of Lemma 4.5**: Since $\hat{\boldsymbol{x}}_\rho$ and $\hat{\boldsymbol{z}}_\rho$ solve the optimization problem RALASSO (4.5), we have,

$$\frac{1}{2}\|\boldsymbol{A}\hat{\boldsymbol{x}}_\rho-\boldsymbol{b}\|_2^2 + \lambda\|\hat{\boldsymbol{z}}_\rho\|_1 + \frac{1}{2}\rho\|\boldsymbol{D}^*\hat{\boldsymbol{x}}_\rho-\hat{\boldsymbol{z}}_\rho\|_2^2 \leq \frac{1}{2}\|\boldsymbol{A}\boldsymbol{x}-\boldsymbol{b}\|_2^2 + \lambda\|\boldsymbol{D}^*\boldsymbol{x}\|_1.$$

Since $\boldsymbol{b} = \boldsymbol{A}\boldsymbol{x}+\boldsymbol{w}$ and $\boldsymbol{h} = \hat{\boldsymbol{x}}_\rho - \boldsymbol{x}$, it follows that

$$\frac{1}{2}\|\boldsymbol{A}\boldsymbol{h}-\boldsymbol{w}\|_2^2 + \lambda\|\hat{\boldsymbol{z}}_\rho\|_1 + \frac{1}{2}\rho\|\boldsymbol{D}^*\hat{\boldsymbol{x}}_\rho-\hat{\boldsymbol{z}}_\rho\|_2^2 \leq \frac{1}{2}\|\boldsymbol{w}\|_2^2 + \lambda\|\boldsymbol{D}^*\boldsymbol{x}\|_1.$$

Expanding and rearranging the terms in the above equation, we get

$$\frac{1}{2}\|\boldsymbol{A}\boldsymbol{h}\|_2^2 + \lambda\|\hat{\boldsymbol{z}}_\rho\|_1 + \frac{1}{2}\rho\|\boldsymbol{D}^*\hat{\boldsymbol{x}}_\rho-\hat{\boldsymbol{z}}_\rho\|_2^2 \leq \mathrm{Re}\langle\boldsymbol{A}\boldsymbol{h},\boldsymbol{w}\rangle + \lambda\|\boldsymbol{D}^*\boldsymbol{x}\|_1,$$

Using (D.4) to replace the terms with $\hat{\boldsymbol{z}}_\rho$, we have

$$\frac{1}{2}\|\boldsymbol{A}\boldsymbol{h}\|_2^2 + \lambda\left\|\boldsymbol{D}^*\hat{\boldsymbol{x}}_\rho - \frac{\lambda}{\rho}\boldsymbol{v}\right\|_1 + \frac{1}{2}\rho\left\|\frac{\lambda}{\rho}\boldsymbol{v}\right\|_2^2 \leq \mathrm{Re}\langle\boldsymbol{A}\boldsymbol{h},\boldsymbol{w}\rangle + \lambda\|\boldsymbol{D}^*\boldsymbol{x}\|_1.$$

Since $\|\boldsymbol{D}^*\hat{\boldsymbol{x}}_\rho - \frac{\lambda}{\rho}\boldsymbol{v}\|_1 \geq \|\boldsymbol{D}^*\hat{\boldsymbol{x}}_\rho\|_1 - \frac{\lambda}{\rho}\|\boldsymbol{v}\|_1$, we have

$$\begin{aligned}
&\frac{1}{2}\|\boldsymbol{A}\boldsymbol{h}\|_2^2 + \lambda\|\boldsymbol{D}^*\hat{\boldsymbol{x}}_\rho\|_1 \\
&\leq \frac{\lambda^2}{\rho}\|\boldsymbol{v}\|_1 - \frac{\lambda^2}{2\rho}\|\boldsymbol{v}\|_2^2 + \mathrm{Re}\langle\boldsymbol{A}\boldsymbol{h},\boldsymbol{w}\rangle + \lambda\|\boldsymbol{D}^*\boldsymbol{x}\|_1 \\
&\leq \frac{\lambda^2 p}{2\rho} + \mathrm{Re}\langle\boldsymbol{A}\boldsymbol{h},\boldsymbol{w}\rangle + \lambda\|\boldsymbol{D}^*\boldsymbol{x}\|_1. \qquad \text{(D.7)}
\end{aligned}$$

The second inequality follows from the fact that $\frac{\lambda^2}{\rho}\|\boldsymbol{v}\|_1 - \frac{\lambda^2}{2\rho}\|\boldsymbol{v}\|_2^2$ is maximized when every element of $\boldsymbol{v} \in \mathbb{R}^p$ is 1. Now, with the assumption that $\boldsymbol{D}$ is a tight frame, we have the following relation:

$$\begin{aligned}\mathrm{Re}\langle \boldsymbol{A}\boldsymbol{h}, \boldsymbol{w}\rangle + \lambda\|\boldsymbol{D}^*\boldsymbol{x}\|_1 =& \mathrm{Re}\langle \boldsymbol{D}^*\boldsymbol{h}, \boldsymbol{D}^*\boldsymbol{A}^*\boldsymbol{w}\rangle + \lambda\|\boldsymbol{D}^*\boldsymbol{x}\|_1 \\ \leq& \|\boldsymbol{D}^*\boldsymbol{h}\|_1\|\boldsymbol{D}^*\boldsymbol{A}^*\boldsymbol{w}\|_\infty + \lambda\|\boldsymbol{D}^*\boldsymbol{x}\|_1.\end{aligned}$$

This inequality follows from the fact that $\mathrm{Re}\langle \boldsymbol{x}, \boldsymbol{y}\rangle \leq |\langle \boldsymbol{x}, \boldsymbol{y}\rangle| \leq \|\boldsymbol{x}\|_1\|\boldsymbol{y}\|_\infty$. Using the assumption that $\|\boldsymbol{D}^*\boldsymbol{A}^*\boldsymbol{w}\|_\infty \leq \frac{\lambda}{2}$, we get

$$\mathrm{Re}\langle \boldsymbol{A}\boldsymbol{h}, \boldsymbol{w}\rangle + \lambda\|\boldsymbol{D}^*\boldsymbol{x}\|_1 \leq \frac{\lambda}{2}\|\boldsymbol{D}^*\boldsymbol{h}\|_1 + \lambda\|\boldsymbol{D}^*\boldsymbol{x}\|_1. \tag{D.8}$$

Applying inequalities (D.7) and (D.8), we have

$$\begin{aligned}\lambda\|\boldsymbol{D}^*\hat{\boldsymbol{x}}_\rho\|_1 \leq& \frac{1}{2}\|\boldsymbol{A}\boldsymbol{h}\|_2^2 + \lambda\|\boldsymbol{D}^*\hat{\boldsymbol{x}}_\rho\|_1 \\ \leq& \frac{\lambda^2}{2\rho}p + \mathrm{Re}\langle \boldsymbol{A}\boldsymbol{h}, \boldsymbol{w}\rangle + \lambda\|\boldsymbol{D}^*\boldsymbol{x}\|_1 \\ \leq& \frac{\lambda^2}{2\rho}p + \frac{\lambda}{2}\|\boldsymbol{D}^*\boldsymbol{h}\|_1 + \lambda\|\boldsymbol{D}^*\boldsymbol{x}\|_1,\end{aligned}$$

which is the same as,

$$\|\boldsymbol{D}^*\hat{\boldsymbol{x}}_\rho\|_1 \leq \frac{\lambda}{2\rho}p + \frac{1}{2}\|\boldsymbol{D}^*\boldsymbol{h}\|_1 + \|\boldsymbol{D}^*\boldsymbol{x}\|_1.$$

Since we have $\boldsymbol{h} = \hat{\boldsymbol{x}}_\rho - \boldsymbol{x}$, it follows that

$$\|\boldsymbol{D}^*\boldsymbol{h} + \boldsymbol{D}^*\boldsymbol{x}\|_1 \leq \frac{\lambda}{2\rho}p + \frac{1}{2}\|\boldsymbol{D}^*\boldsymbol{h}\|_1 + \|\boldsymbol{D}^*\boldsymbol{x}\|_1,$$

and hence

$$\begin{aligned}&\|\boldsymbol{D}_{\mathcal{T}}^*\boldsymbol{h} + \boldsymbol{D}_{\mathcal{T}}^*\boldsymbol{x}\|_1 + \|\boldsymbol{D}_{\mathcal{T}^c}^*\boldsymbol{h} + \boldsymbol{D}_{\mathcal{T}^c}^*\boldsymbol{x}\|_1 \\ &\leq \frac{\lambda}{2\rho}p + \frac{1}{2}\|\boldsymbol{D}_{\mathcal{T}}^*\boldsymbol{h}\|_1 + \frac{1}{2}\|\boldsymbol{D}_{\mathcal{T}^c}^*\boldsymbol{h}\|_1 + \|\boldsymbol{D}_{\mathcal{T}}^*\boldsymbol{x}\|_1 + \|\boldsymbol{D}_{\mathcal{T}^c}^*\boldsymbol{x}\|_1.\end{aligned}$$

Applying the triangle inequality to the left handside of above inequality, we results in

$$
\begin{aligned}
&- \|\boldsymbol{D}_{\mathcal{T}}^{*}\boldsymbol{h}\|_1 + \|\boldsymbol{D}_{\mathcal{T}}^{*}\boldsymbol{x}\|_1 + \|\boldsymbol{D}_{\mathcal{T}^c}^{*}\boldsymbol{h}\|_1 - \|\boldsymbol{D}_{\mathcal{T}^c}^{*}\boldsymbol{x}\|_1 \\
&\leq \frac{\lambda}{2\rho}p + \frac{1}{2}\|\boldsymbol{D}_{\mathcal{T}}^{*}\boldsymbol{h}\|_1 + \frac{1}{2}\|\boldsymbol{D}_{\mathcal{T}^c}^{*}\boldsymbol{h}\|_1 + \|\boldsymbol{D}_{\mathcal{T}}^{*}\boldsymbol{x}\|_1 + \|\boldsymbol{D}_{\mathcal{T}^c}^{*}\boldsymbol{x}\|_1.
\end{aligned}
$$

After rearranging the terms, we have the following cone constraint,

$$
\|\boldsymbol{D}_{\mathcal{T}^c}^{*}\boldsymbol{h}\|_1 \leq \frac{\lambda}{\rho}p + 3\|\boldsymbol{D}_{\mathcal{T}}^{*}\boldsymbol{h}\|_1 + 4\|\boldsymbol{D}_{\mathcal{T}^c}^{*}\boldsymbol{x}\|_1. \tag{D.9}
$$

# Vita

Zhao Tan

**Degrees**

Ph.D., Electrical Engineering, Washington University in St. Louis, Missouri, USA, May 2015
M.S., Electrical Engineering, Washington University in St. Louis, Missouri, USA, December 2013
B.S., Electrical Engineering, Fudan University, Shanghai, China, June 2010

**Professional Societies**

The Institute of Electrical and Electronics Engineers (IEEE)
IEEE Signal Processing Society

**Publications**

Book Chapter:

**Z. Tan**, P. Yang, and A. Nehorai, "An Optimal and Distributed Control Strategy for Charging Plug-in Electrical Vehicles in the Smart Grid," in *Plug-In Electric Vehicles in Smart Grid: Management and Control Strategies*, (Ch. 4, pp. 79- 106, S. Rajakaruna, Ed.) Springer, Berlin Heidelberg, 2015.

Journal Publications:

**Z. Tan**, Y. C. Eldar and A. Nehorai,"Direction of arrival estimation with co-prime arrays: a super resolution viewpoint," *IEEE Trans. Signal Processing*, Vol. 62, pp. 5565-5576, Nov. 2014.

**Z. Tan**, P. Yang, and A. Nehorai, "Joint sparse recovery method for compressed sensing with structured dictionary mismatch," *IEEE Trans. Signal Processing*, Vol. 62, pp. 4997-5008, Oct. 2014.

**Z. Tan**,Y. C. Eldar, A. Beck and A. Nehorai,"Smoothing and decomposition for analysis sparse recovery," *IEEE Trans. Signal Processing*, vol. 62, pp. 1762-1774, April 2014.

**Z. Tan**, P. Yang, and A. Nehorai, "An optimal and distributed demand response strategy with electric vehicles in the smart grid," *IEEE Trans. on Smart Grid*, Vol 5, No. 2, pp. 861-869, Mar. 2014.

**Z. Tan**, and A. Nehorai,"Sparse direction of arrival estimation using co-prime arrays with off-grid targets," *IEEE Signal Processing Letters*, vol. 21, no. 1, pp. 26-29, Jan. 2014.

P. Yang, **Z. Tan**, A. Wiesel, and A. Nehorai, "Placement of phasor measurement units considering imperfect time synchronization," *IEEE Trans. on Power Delivery*, Vol 30, No. 2, pp. 914-922, Apr. 2015..

P. Yang, **Z. Tan**, A. Wiesel, and A. Nehorai, "Power system state estimation using PMUs with imperfect synchronization," *IEEE Trans. on Power Systems*, Vol 28, No. 4, pp. 4162-4172, Nov. 2013.

Y. Chen*, C. Moran*, **Z. Tan***, A. Wooten*, J. O'Sullivan,"Robust analysis of multiplexed SERS microscopy of Ag nanocubes using an alternating minimization algorithm," *Journal of Raman Spectroscopy*, vol. 44, issue 5, May 2013.

Conference Publications:

**Z. Tan**, Y. C. Eldar and A. Nehorai,"Continuous Sparse Recovery for Direction of Arrival Estimation with Co-prime Arrays," *Sensor Array and Multichannel Signal Processing (SAM), 2014 IEEE 8th International Workshop on*, pp.393 - 396, 22-25 June 2014.

**Z. Tan**, P. Yang, and A. Nehorai, "Distributed demand response for plug-in electrical vehicles in the smart grid," in Proc. *5th IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing(CAMSAP)*, Saint Martin, Dec. 2013.

**Z. Tan**, P. Yang, and A. Nehorai, "Joint-sparse recovery in compressed sensing with dictionary mismatch," in Proc. *5th IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing(CAMSAP)*, Saint Martin, Dec. 2013.

P. Yang, **Z. Tan**, A. Wiesel, and A. Nehorai, "Performance bounds and sensor placement for state estimation using PMUs with phase mismatch," in Proc. *IEEE Power and Energy Society (PES) General Meeting*, Vancouver BC, Canada, July, 2013.

**Z. Tan**, P. Yang, and A. Nehorai, "Sparse MIMO radar with phase mismatch" in Proc. *38th IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Vancouver BC, Canada, May, 2013.

P. Yang, **Z. Tan**, A. Wiesel, and A. Nehorai, "State estimation with consideration of PMU phase mismatch for smart grids," in Proc. *IEEE Innovative Smart Grid Technologies Conference (ISGT)*, Washington D.C., Feb, 2013.

May 2015