Fall 2017

Convex and non-convex optimizations for recovering structured data: algorithms and analysis

Myung Cho
University of Iowa

CONVEX AND NON-CONVEX OPTIMIZATIONS FOR RECOVERING STRUCTURED DATA: ALGORITHMS AND ANALYSIS

by

Myung Cho

A thesis submitted in partial fulfillment of the
requirements for the Doctor of Philosophy
degree in Electrical and Computer Engineering
in the Graduate College of
The University of Iowa

December 2017

Thesis supervisor: Professor Weiyu Xu

Graduate College
The University of Iowa
Iowa City, Iowa

CERTIFICATE OF APPROVAL

PH.D. THESIS

This is to certify that the Ph.D. thesis of

Myung Cho

has been approved by the Examining Committee for the thesis requirement for the Doctor of Philosophy degree in Electrical and Computer Engineering at the December 2017 graduation.

Thesis committee: _____
Weiyu Xu, Thesis Supervisor

_____
Soura Dasgupta

_____
Xiaodong Wu

_____
Mathews Jacob

_____
Qihang Lin

To my wife and son

ACKNOWLEDGEMENTS

I sincerely thank my advisor, Professor Weiyu Xu, for his limitless support, enthusiasm and inspiration throughout my Ph.D. study. I would like to express my thanks to Professor Xiaodong Wu in the Department of ECE, Dr. Ryan T. Flynn and Dr. Yusung Kim in the Department of Radiation Oncology, University of Iowa Hospital and Clinics for introducing me to their broad view of research in the rotating-shield brachytherapy. Additionally, I thank Professor Soura Dasgupta and Professor Mathews Jacob in the Department of ECE, and Professor Qihang Lin in the department of Business for their great support as committee members.

I would like to give my special thanks to my family who support me endlessly during my study. Without their support, I couldn't even start and finish my Ph.D. study.

ABSTRACT

Optimization theories and algorithms are used to efficiently find optimal solutions under constraints. In the era of "Big Data", the amount of data is skyrocketing, and this overwhelms conventional techniques used to solve large scale and distributed optimization problems. By taking advantage of structural information in data representations, this thesis offers convex and non-convex optimization solutions to various large scale optimization problems such as super-resolution, sparse signal processing, hypothesis testing, machine learning, and treatment planning for brachytherapy.

Super-resolution: Super-resolution aims to recover a signal expressed as a sum of a few Dirac delta functions in the time domain from measurements in the frequency domain. The challenge is that the possible locations of the delta functions are in the continuous domain [0,1). To enhance recovery performance, we considered deterministic and probabilistic prior information for the locations of the delta functions and provided novel semidefinite programming formulations under the information. We also proposed block iterative reweighted methods to improve recovery performance without prior information. We further considered phaseless measurements, motivated by applications in optic microscopy and x-ray crystallography. By using the lifting method and introducing the squared atomic norm minimization, we can achieve super-resolution using only low frequency magnitude information. Finally, we proposed non-convex algorithms using structured matrix completion.

Sparse signal processing: ℓ_1 minimization is well known for promoting sparse

structures in recovered signals. The Null Space Condition (NSC) for ℓ_1 minimization is a necessary and sufficient condition on sensing matrices such that a sparse signal can be uniquely recovered via ℓ_1 minimization. However, verifying NSC is a non-convex problem and known to be NP-hard. We proposed enumeration-based polynomial-time algorithms to provide performance bounds on NSC, and efficient algorithms to verify NSC precisely by using the branch and bound method.

Hypothesis testing: Recovering statistical structures of random variables is important in some applications such as cognitive radio. Our goal is distinguishing two different types of random variables among $n >> 1$ random variables. Distinguishing them via experiments for each random variable one by one takes lots of time and efforts. Hence, we proposed hypothesis testing using mixed measurements to reduce sample complexity. We also designed efficient algorithms to solve large scale problems.

Machine learning: When feature data are stored in a tree structured network having time delay in communication, quickly finding an optimal solution to the regularized loss minimization is challenging. In this scenario, we studied a communication-efficient stochastic dual coordinate ascent and its convergence analysis.

Treatment planning: In the Rotating-Shield Brachytherapy (RSBT) for cancer treatment, there is a compelling need to quickly obtain optimal treatment plans to enable clinical usage. However, due to the degree of freedom in RSBT, finding optimal treatment planning is difficult. For this, we designed a first order dose optimization method based on the alternating direction method of multipliers, and reduced the execution time around 18 times compared to the previous research.

PUBLIC ABSTRACT

Optimization theories and algorithms are used to efficiently find optimal solutions under constraints. In the era of "Big Data", the amount of data is skyrocketing, and this overwhelms conventional techniques used to solve large scale and distributed optimization problems. By taking advantage of structural information in data representations, this thesis offers convex and non-convex optimization solutions to various large scale optimization problems such as super-resolution, sparse signal processing, hypothesis testing, machine learning, and treatment planning for brachytherapy.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1
INTRODUCTION

1.1 Motivation

In engineering and science, optimization algorithms and theories play a significant role in finding optimal solutions to problems that have lots of constraints. Mathematicians, economists, and engineers in various fields have studied these optimization problems extensively. For example, in signal processing, researchers have investigated algorithms to recover signals and images with limited measurements and observations. In economics, investors have tried to maximize profit from limited financial budget.

Thanks to improved sensing technologies, we can easily collect data and measurements from various sensors and devices. Thus, the amount of data that we should deal with in optimization problems is rapidly increasing. Also, a lot of emerging applications such as smart grid, electric vehicle charge scheduling, transportation networks, and Internet of things (IoT) require large scale optimization due to their nature of broad connectivity and huge number of objects. In addition to large problem sizes, recent optimization problems are different from traditional optimization problems in other aspects such as distributed data and new data structures. These differences make conventional solvers inapplicable to these large scale distributed optimization problems. Thus, the demand for solutions to large scale optimization problems is ever growing and opening up a variety of new research possibilities. It necessitates

developing optimization algorithms and theories that can deal with large scale and distributed optimization problems.

1.2 Background

Optimization theory is a framework for selecting the best solution from feasible options. An optimization problem can be mathematically formulated as follows:

$$\begin{aligned} &\underset{x\in\mathbb{R}^n}{\text{minimize}} \;\; f(x) \\ &\text{subject to} \;\; x \in \mathcal{X}, \end{aligned} \tag{1.1}$$

where the vector x is a variable, $f : \mathbb{R}^n \to \mathbb{R}$ is an objective function, and $\mathcal{X}$ is a feasible set. The following subsections describe different types of optimization problems considered in this thesis.

1.2.1 Convex optimization

The definition of a convex set in [20] is stated as follows: A set $\mathcal{X}$ is convex if and only if the line segment between any two points in $\mathcal{X}$ lies in $\mathcal{X}$; namely, if for any $x, y \in \mathcal{X}$ and $\alpha \in [0,1]$, we have

$$\alpha x + (1-\alpha)y \in \mathcal{X}. \tag{1.2}$$

A function $f : \mathbb{R}^n \to \mathbb{R}$ is convex if the domain of f, denoted by $dom(f)$, is a convex set, and for all $x, y \in dom(f)$, and $\alpha \in [0,1]$, we have

$$f(\alpha x + (1-\alpha)y) \leq \alpha f(x) + (1-\alpha)f(y). \tag{1.3}$$

A convex optimization problem has a convex objective function and a convex feasible set.

Convex optimization plays a significant role in mathematical optimization because it has several advantages. The first advantage is that a local optimal solution is also a global optimal solution in convex optimization problems, so that we can safely terminate optimization algorithms when we find a local optimal solution. Secondly, for convex optimization problems, theoretical convergence of gradient descant is guaranteed. Hence, we can predict how fast we can obtain a solution. Finally, a (primal) convex optimization problem can have its dual problem whose optimal value lower-bounds the optimal value of the primal problem. When strong duality holds, the primal and dual optimal values become the same. Thus, if the primal problem is hard to solve, we can solve its dual problem to obtain an optimal solution, or vice versa. This condition can also be used for stopping criteria of iterative algorithms. In the following subsections, we introduce different types of convex optimization problems.

1.2.1.1 Linear programming

Linear programming is an important class of convex optimization problem, which has linear objective function, linear equality and linear inequality constraints. Mathematically, we can state a linear programming as follows:

$$\begin{aligned} &\underset{x \in \mathbb{R}^n}{\text{minimize}} \;\; c^T x \\ &\text{subject to} \;\; a_i^T x \leq b_i, \; i = 1, 2, ..., m, \end{aligned} \tag{1.4}$$

where c, a_1, ..., $a_m \in \mathbb{R}^n$ are given vectors and b_i, $i = 1, 2, ..., m$, are given constants. Its feasible region is expressed as a convex polytope, which is the intersection of half

spaces. We denote the objective function $c^T x$ as $f(x)$. For any $\alpha, \beta \in \mathbb{R}$, we have

$$f(\alpha x + \beta y) = \alpha f(x) + \beta f(y). \tag{1.5}$$

Note that unlike a generic convex function in (1.3), (1.5) has equality.

Linear programming problems have found applications in various fields such as economics, engineering, power and energy industry, etc. Even though we have no close-form solutions for generic linear programming, lots of efficient algorithms to solve linear programming problems ranging from Danzig simplex algorithm to Karmarkar's algorithm are introduced in literature [20].

1.2.1.2 Semidefinite programming

Semidefinite programming is another special convex optimization problem minimizing a linear objective function over the feasible set expressed as the intersection of the cone of positive semidefinite matrices with an affine space. Mathematically a semidefinite programming is stated as follows:

$$\begin{aligned} &\underset{x \in \mathbb{R}^n}{\text{minimize}} \;\; c^T x \\ &\text{subject to} \;\; x_1 F_1 + x_2 F_2 + ... + x_n F_n + G \preceq 0, \\ &\qquad\qquad Ax = b, \end{aligned} \tag{1.6}$$

where x_i is the i-th element of the vector x, F_1, $F_2, ...,$ F_n, G are $m \times m$ symmetric matrices, $A \in \mathbb{R}^{p \times n}$ and $b \in \mathbb{R}^p$. It is noteworthy that if F_1, $F_2, ...,$ F_n, G are diagonal matrices, then semidefinite programming (1.6) becomes linear programming (1.4).

There are a variety of algorithms to solve semidefinite programming problems. Interior point methods are well known and implemented in general solvers ranging

from SDPT3 to SeDuMi [91]. In Chapter 3, we consider semidefinite programming formulations for solving super-resolution problems.

1.2.1.3 First-order method

In situations where problems are extremely large, more efficient and faster methods to solve optimization problems are highly desired. Fast algorithms based on the first-order method are receiving lots of attention due to their speed and practical uses. Hence, developing fast algorithms based on the first-order method has been actively researched.

One powerful algorithm to solve convex optimization problems is the Alternating Direction Method of Multipliers (ADMM). It is also considered a good candidate solver for large scale optimization problems due to its convergence to an optimal solution and applicability to various modern large scale applications ranging from machine learning to computer vision [19]. The optimization problem that we consider in ADMM is stated as follows:

$$\begin{aligned} &\underset{x\in\mathbb{R}^n,\, z\in\mathbb{R}^l}{\text{minimize}} \;\; f(x)+g(z) \\ &\text{subject to} \;\; Ax+Bz=c, \end{aligned} \tag{1.7}$$

where $f(x)$ and $g(z)$ are convex functions, $A \in \mathbb{R}^{m\times n}$, $B \in \mathbb{R}^{m\times l}$, and $c \in \mathbb{R}^m$. The strategy of ADMM is using the Lagrange multiplier Λ for dual variable of the equality constraint $Ax+Bz=c$, and adding a quadratic penalty term for the equality constraint

as follows:

$$L_a(x, z, \Lambda) = f(x) + h(z) + \langle \Lambda, c - Ax - Bz \rangle + \frac{\rho}{2}||c - Ax - Bz||_2^2, \tag{1.8}$$

where $\rho > 0$ is a penalty parameter. Here the function $L_a(x, z, \Lambda)$ is called the *augmented Lagrangian function*. By considering the augmented Lagrangian function, ADMM searches for the optimal solution without maintaining feasibility at all iterations. ADMM minimizes this augmented Lagrangian function over x, z, and Λ iteratively until it converges.

The rate of convergence shows how fast the intermediate result of an iterative algorithm goes to the optimal solution by considering the distance at each iteration between the intermediate result and the optimal value. Hence, it is simply represented in big-O notation such as $O(\frac{1}{r})$ for sublinear, and $O(c^r)$ for linear convergence, where r is the number of iteration and c is a constant. The convergence of ADMM has been well studied under the various conditions. Several studies have shown that the convergence rate of ADMM is sublinear such as $O(\frac{1}{r})$ or $O(\frac{1}{r^2})$ at accelerated version [62, 94]. In particular, when the objective function is strongly convex, it has been shown that the convergence of ADMM is linear [62, 90]. In Chapter 6, we consider the proximal graph solver, which is a convex optimization solver based on ADMM, for large scale cancer treatment planning in brachytherapy.

Another algorithm to solve large scale convex optimization problems is gradient descent. The gradient descent is a conventional iterative method to solve an optimization problem (1.1). In gradient descent, the $(r+1)$-th value for x is obtained

by calculating the following updating equation:

$$x^{(r+1)} = x^{(r)} - \alpha^{(r)} \bigtriangledown f(x^{(r)}), \tag{1.9}$$

where the super-script represents the iteration number, α is step size, and $\bigtriangledown f(x^{(r)})$ is the gradient value of the objective function f at the current point $x^{(r)}$. In order to improve the performance of the conventional gradient descent, people have developed various enhanced gradient decent algorithms such as stochastic gradient descent, proximal gradient descent, primal-dual gradient methods, and coordinate gradient descent methods [20, 143, 162]. Especially, accelerate and fast gradient descent methods and their analysis [10, 143] have been studied extensively, and widely used in various applications such as computer vision, machine learning, signal processing and communications [162]. In Chapter 5, we study the first-order method to solve extremely large scale optimization problems in a distributed network system.

1.2.2 Non-convex optimization

In contrast to convex optimization problems, a non-convex optimization problem is known to be NP-hard in most cases. A non-convex optimization problem has a non-convex objective function $f(x)$ or a non-convex feasible set $\mathcal{X}$ in (1.1). Since non-convex problems can have lots of local minimums, figuring out a global solution from lots of local minimums is quite challenging. Due to the difficulty in solving non-convex optimization problems, people approached non-convex problems in the following ways: relaxing non-convex problems to convex problems, using convex optimization to find upper or lower bounds on non-convex problems, and further

using the bounds to find a global optimal solution. Since some practical problems are expressed as non-convex problems, research on various non-convex optimization methods has been widely conducted [144]. In the following subsections, we describe the approaches to solve non-convex optimization problems in detail.

1.2.2.1 Relaxation to convex optimization

In relaxation, a non-convex objective function is replaced to its close convex function, and the non-convex constraints are replaced to loose convex constraints, e.g., convex hull. By relaxing the objective function or the constraints of the non-convex optimization in various ways, we can have several different relaxed convex optimization problems. Even though a solution to a non-convex optimization problem can be different from a solution to its relaxed convex optimization problems, the relaxation can provide several advantages ranging from computational efficiency to an approximate solution to the non-convex optimization. Since the solution to the non-convex optimization can be bounded by the solution to the convex optimization, we can use this bound to find a global solution to the non-convex optimization.

1.2.2.2 Strategy to find a global solution

Non-convex function can have lots of local minimums. Among them, finding a global solution is computationally intractable. A simple method to find a global solution is an exhaustive search method. However, as the size of the feasible set increases, the exhaustive search method becomes impractical. In order to reduce the search range, using lower bounds and upper bounds on a global solution is used in

the branch and bound method for non-convex optimization [57]. By reducing a global upper bound and increasing a global lower bound, we can finally find a global solution efficiently.

1.2.2.3 Algorithms to solve non-convex optimization problems

There are non-convex algorithms to solve non-convex optimization problems. Especially, in low rank matrix completion problems, rank of a matrix is considered as an objective function or a constraint. Instead of relaxing the rank of a matrix to the nuclear norm of a matrix or the trace of a matrix, directly using the rank of a matrix is also possible by conducting the singular value decomposition of the matrix at each iteration. In Section 3.4, we use this non-convex algorithm for super-resolution problems. In addition, gradient descent can also be used to find local minimum of a non-convex function in non-convex optimization problems [87].

1.2.3 Using structural information in data

Compressed sensing is an efficient signal processing technique used to recover a sparse signal from limited observation data. Here, a sparse signal represents a signal has only a few nonzero elements. Research on compressed sensing has shown that when a signal of interest is sparse, the signal can be recovered from fewer samples than the Nyquist-Shannon sampling theorem requires. Hence, compressed sensing can reduce the required sampling rate for recovery of the original signal. Because of this advantage, compressed sensing is used in various applications such as medical imaging, face recognition, and hologram [77].

One popular optimization problem considered in compressed sensing, called ℓ_1 minimization, is stated as follows:

$$\begin{aligned} &\underset{x\in\mathbb{R}^n}{\text{minimize}} \ \|x\|_1 \\ &\text{subject to } \ x \in \mathcal{X}, \end{aligned} \tag{1.10}$$

where $\|x\|_1$ is ℓ_1 norm of x, which is the sum of absolute values of elements of x, i.e., $\|x\|_1 \triangleq \sum_{i=1}^{n} |x_i|$, and $\mathcal{X}$ is a convex set. In compressed sensing, it is well known that we can obtain the sparsest solution by solving ℓ_1 minimization problem under the Null Space Condition (NSC) [52, 55, 110, 152, 171] and the Restricted Isometry Property (RIP) [31, 171]. Extensive research has been dedicated to designing efficient algorithms to solve ℓ_1 minimization [15, 32, 65, 174].

Optimization using structural information can be further extended to matrix completion problem, which involves finding a low rank matrix from a few randomly observed matrix elements. For this problem, the authors in [28, 151] propose nuclear norm minimization relaxations. Here, the nuclear norm $\|X\|_*$ is represented as the sum of singular values $\rho_i(X)$ of a matrix X, i.e., $\|X\|_* \triangleq \sum_{i=1}^{k} \rho_i(X)$, $rank(X) = k$. By solving this nuclear norm minimization, one can obtain a unique low rank matrix. Hence, the nuclear norm minimization method can be successfully applied to recommendation systems, e.g., the Netflix problem, by providing a solution for low-rank matrix completion. Efficient algorithms to solve matrix completion problems were also studied in [22, 113, 128, 139].

On a similar track, line spectral problems [45, 51, 98, 138, 168] and phase retrieval problems [24], which involve extracting and recovering frequency or phase

information, are being actively researched. In particular, atomic norm minimization has benn proposed to exploit structural information in these problems.

In performing optimization, there are other types of structural information in data such as block sparse vector, block diagonal matrix, Hankel matrix, Toeplitz matrix, and full rank matrix. Designing efficient algorithms to solve optimization problems with structural information is an on-going research topic. In Chapters 2, 3, 4, 5, and 6, we consider optimization problems using structural information in detail.

1.3 Challenges in Large Scale Optimization

1. **Size of problem**: When the size of the problem is big, large memory space is required to store large size of intermediate values or given data set. In addition, calculating the derivative information in full dimension can be considerably slow. This makes it difficult to check the computational progress in the middle of calculation. Thus, the size of problem can be a constraint in solving large scale problems.

2. **Data accessibility**: Due to cheap data acquisition from sensors, data are collected broadly, distributively and simultaneously over all the time indices and places. Transmitting those collected big data from one point to another are costly. Hence, processing all the data in a central point becomes difficult.

3. **Incomplete data**: Since data are collected independently over time and space, the data describing a optimization problem can be incomplete and unavailable during a certain interval or at specific place. This happens frequently in dis-

tributed networks. Waiting for entire data to be corrected makes the calculation of the problem inefficient and slow. Also, the available data can be contaminated by noise and outliers.

4. **Limit of computational power**: In solving a large scale optimization problem, even a simple full-dimensional vector operation can be very expensive such as calculating gradient value. Hence, reducing computational burden in full-dimensional vector operations is required.

5. **Limit of time**: Practical applications require fast computational responses. The algorithms should be fast in order to be used in large scale optimization problems.

1.4 Applications

Optimization techniques can play a significant role in various applications as follows:

1. **Medical treatment planning**: The Rotating-Shield Brachytherapy (RSBT) treatment planning is one of biomedical techniques for cancer treatment, which delivers directionally modulate radiation to a target organ by rotating radiation-attenuating shield around a brachytherapy source. The major problem in RSBT treatment planning is determining optimal dose exposure time and locations for the target organ without exceeding the tolerance doses of non-target organ in an optimized fashion. For example, for prostrate treatment, there are many possible needle locations that brachytherapy source can be delivered in. For

each needle locations, there are also many positions for dose exposure, which is called dwell positions. At each dwell position, the target organ gets exposed to radiation by rotating a shield around a brachytherapy source in predetermined degree, e.g., 20 degrees, at each time, and it is needed to cover whole 360 degrees. The goal here is finding optimal treatment time and needle locations from millions of possible treatment planning. Hence, this is one of applications in which a large scale optimization technique is needed.

2. **Machine learning**: The amount of collected data is extremely huge and broad, because data acquisition and regeneration become cheap and easy. People call this extremely amount of data as *big data*. Various global companies such as Google, Amazon, and Facebook are trying to improve their service and increase their incomes by using those massive data. One good example for this is Google Flu Trends, which is a web service that Google provides. In the service, Google provides the anticipated time of flu outbreak by using search engine query data, and reported that 95% prediction intervals were indicated in [89]. From this simple example, massive data can play an essential role in obtaining useful information. Machine learning is designing algorithms that can extract useful information from massive data set. Since the size of data set is increased and huge, the operation time in machine learning operations to extract useful information is also increased and becomes a bottle-neck. Hence, the fast optimization technique that can handle large scale optimization problems is required.

3. **Network system optimization**: There are various network systems such as smart Grid, wireless sensor networks, multi-agent systems, networked control systems, and communication networks. The characteristic that those network systems share in common is a distributed system. For example, in smart grid, power generators are distributed. Wireless sensor networks have distributed sensors, and communication networks have base stations and users in different places. In the network framework, the distributed elements such as power generator, sensor, base station, and user are represented as nodes, and the connection between two nodes is expressed as an edge in the network. Since the environment and situation that each node has are different, each node can have shortage and excess of power or information that each system considers. Hence, there is a room for improvement of a whole system by sharing the storage and excess at one node with other nodes in optimal manner. Additionally, the current network systems become more complex than those in the past. Hence, the size of optimization problems in network systems is dramatically increased.

4. **Image processing**: Image processing is a prominent field in which optimization techniques using structure information are widely used. Especially, medical image applications such as MRI and CT scan require high quality images to diagnosis disease clearly. However, it takes a large amount of time to take those high quality images from MRI and CT scan. Hence, there are substantial efforts to reduce the time by using optimization technique with structure information. For high quality images, researchers are focusing on the methods to obtain high

resolution information from samples in low resolution by using optimization theory. This is known as super-resolution. Also, images can be easily contaminated by noise, which makes images hard to be recognized. The optimization technique comes into play for image denoising to reduce noise factors.

5. **Financial engineering**: Financial engineering is a multidisciplinary field that considers engineering methods and tools including optimization theory and technique to solve financial problems. In financial engineering, optimization technique can play an important role in maximizing incomes within given restrictions, and providing optimal financial strategies to maximize profits. Moreover, as it is shown through the subprime mortgage crisis happened in U.S., and the Asian financial crisis, the current financial problems become huge and complicated, because of connections in investments. Therefore, large scale optimization technique is required in this field.

1.5 Organization and Contribution

The remainder of this thesis is organized as follows. We provide our contributions in each chapter as well.

- **Chapter 2**

  We consider the problem of verifying the null space condition, which is a sufficient and necessary condition to guarantee a unique sparse solution from ℓ_1 minimization. It is a non-convex optimization problem. For this problem, we propose fast algorithms to verify the null space condition with upper bound

as well as enhanced algorithms to verify the null space condition precisely. In application-wise, we show that our algorithms can be used for checking the robustness of networks in defective edge detection aspect. Here, the robustness of networks is expressed as the maximum number of defective edges in networks which can be uniquely found at the time of detection via ℓ_1 minimization with delay information obtained from random walk.

- **Chapter 3**

  In this chapter, we deal with a super-resolution problem. Our goal here is to figure out time signal information, which can take any value between 0 and 1, from as few frequency samples as possible. Hence, the size of dimension in time is infinite. We propose the super-resolution method dealing with various prior information, and then introduce the block iterative reweighted algorithms to deal with the case without prior information. After that, we consider the super-resolution problem with phaseless measurement, called phaseless super-resolution. Finally, we propose fast algorithms to solve the super-resolution problem by using structured matrix completion.

- **Chapter 4**

  This chapter describes a signal detection problem in the statistic signal processing. We aim to detect a few abnormal random variables out of lots of normal random variables. The abnormal and normal random variable follow different probability distributions. We use mixed observations of random variables in-

stead of single observation of a random variable at each experiment, which we call it as *compressed hypothesis testing*. We propose Neyman-Pearson testing and maximum likelihood testing methods for optimal detection algorithms as well as efficient algorithms for the large scale detection problems. We show that we can reduce the number of observations in detection of abnormal random variables by using mixed observations.

- **Chapter 5**

  This chapter describes large scale optimization problems for big data. In this problem, we assume that the dimension of the problem is extremely huge and data are distributed to lots of local workers. Hence, computing a simple gradient calculation in full dimension is even expensive or impossible due to the nature of distribution and huge size of data. In this setting, we study distributed dual coordinate ascent in a tree network system, which is generalized network system from a star network having a center computer and local workers connected to the center, to handle extremely large scale problems in distributed updating manner. Furthermore, we consider the case when the network has time delay constraint and provide optimization problem for the optimal number of local iterations.

- **Chapter 6**

  The Rotating-Shield BrachyTherapy (RSBT) is a practical biomedical application for cancer treatment. For practical use in a clinic, calculating optimal

treatment plans for RSBT is required. However, RSBT cancer treatment planning has a high degree of freedom due to lots of variables such as radiation exposure duration time in each radiation source carried by a needle, number of needles used in treatment, and needle locations, which makes the problem difficult. The goal of the research on this practical application is providing a fast optimization solution for optimal cancer treatment in RSBT. For this purpose, we consider a fast computational method, based on the proximal graph solver (POGS) - a convex optimization solver using the alternating direction method of multipliers (ADMM).

- **Chapter 7**

  In this chapter, we introduce future research directions.

CHAPTER 2
COMPUTABLE PERFORMANCE GUARANTEES FOR COMPRESSED SENSING MATRICES

2.1 Introduction

Compressed sensing is an efficient signal processing technique to recover a sparse signal from fewer samples than required by the Nyquist-Shannon theorem, reducing time and energy spent in sampling operation. These advantages make compressed sensing attractive in various signal processing areas [77].

In compressed sensing, we are interested in recovering the sparsest vector $x \in \mathbb{R}^n$ that satisfies the underdetermined equation $y = Ax$. Here $\mathbb{R}$ is the set of real numbers, $A \in \mathbb{R}^{m \times n}$, $m < n$, is a sensing matrix, and $y \in \mathbb{R}^m$ is the observation or measurement data. This is posed as an ℓ_0 minimization problem:

$$\begin{aligned} &\text{minimize } \|x\|_0 \\ &\text{subject to } y = Ax, \end{aligned} \tag{2.1}$$

where $\|x\|_0$ is the number of non-zero elements in vector x. The ℓ_0 minimization is an NP-hard problem. Therefore, we often relax (2.1) to its closest convex approximation - the ℓ_1 minimization problem:

$$\begin{aligned} &\text{minimize } \|x\|_1 \\ &\text{subject to } y = Ax. \end{aligned} \tag{2.2}$$

It has been shown that the optimal solution of ℓ_0 minimization can be obtained by solving ℓ_1 minimization under certain conditions (e.g. Restricted Isometry Property

or RIP) [29–31, 60, 64]. For random sensing matrices, these conditions hold with high probability. We note that RIP is a sufficient condition for sparse recovery [171].

A necessary and sufficient condition under which a k-sparse signal x, ($k \ll n$) can be uniquely obtained via ℓ_1 minimization is Null Space Condition (NSC) [55, 60, 110]. A matrix A satisfies NSC for a positive integer k if

$$\|z_K\|_1 < \|z_{\overline{K}}\|_1 \tag{2.3}$$

holds true for all $z \in \{z : \ Az = 0, z \neq 0\}$ and for all subsets $K \subseteq \{1, 2, ..., n\}$ with $|K| \leq k$. Here K is an index set, $|K|$ is the cardinality of K, z_K is the part of the vector z over the index set K, and $\overline{K}$ is the complement of K. NSC is related to the proportion parameter α_k defined as

$$\alpha_k \triangleq \underset{\{z:\, Az=0,\, z\neq 0\}}{\text{maximize}} \ \underset{\{K:\, |K|\leq k\}}{\text{maximize}} \ \frac{\|z_K\|_1}{\|z\|_1}. \tag{2.4}$$

The α_k is the optimal value of the following optimization problem:

$$\begin{aligned} &\underset{z,\{K:\, |K|\leq k\}}{\text{maximize}} \ \|z_K\|_1 \\ &\text{subject to } \ \|z\|_1 \leq 1, \ Az = 0, \end{aligned} \tag{2.5}$$

where K is a subset of $\{1, 2, \ldots, n\}$ with cardinality at most k. The matrix A satisfies NSC for a positive integer k if and only if $\alpha_k < \frac{1}{2}$. Equivalently, NSC can be verified by computing or estimating α_k. The role of α_k is also important in recovery of an approximately sparse signal x via ℓ_1 minimization where a smaller α_k implies more robustness [55, 110, 190].

We are interested in computing α_k and, especially, finding the maximum k for which $\alpha_k < \frac{1}{2}$. However, computing α_k to verify NSC is extremely expensive and was

reported in [171] to be NP-hard. Due to the challenges in computing α_k, verifying NSC explicitly for deterministic sensing matrices remains a relatively unexamined research area. In [60,110,120,169], convex relaxations were used to establish the upper or lower bounds of α_k (or other parameters related to α_k) instead of computing the exact α_k. While [60, 120] proposed semidefinite programming based methods, [110, 169] suggested linear programming relaxations to obtain the upper and lower bounds of α_k. For both methods, computable performance guarantees on sparse signal recovery were reported via bounding α_k. However, these bounds of α_k could only verify NSC with $k = O(\sqrt{n})$, even though theoretically k can grow linearly with n.

Our work drastically departs from these prior methods [60, 110, 120, 169] that provide only the upper and lower bounds. In our solution, we propose the *pick-l-element* algorithms ($1 \leq l < k$), which compute upper bounds of α_k in polynomial time. Subsequently, we leverage on these algorithms to develop the *tree search algorithm* - a new method to compute the exact α_k by significantly reducing computational complexity of an exhaustive search method. This algorithm offers a way to control a smooth tradeoff between complexity and accuracy of the computations. In the conference precursor to this paper, we had introduced *Sandwiching Algorithm (SWA)* [52], which employs a branch-and-bound method. Although SWA can also be used to calculate the exact α_k, it has a disadvantage of greater memory usage than TSA. On the other hand, TSA provides memory and performance benefits for high-dimensional matrices (e.g., up to size $\sim 6000 \times 6000$).

It is noteworthy that our methods are different from RIP or the neighborly

polytope framework for analyzing the sparse recovery capability of random sensing matrices. For example, prior works such as [64, 69] employ the neighborly polytope to predict theoretical lower bounds on recoverable sparsity k for a randomly chosen Gaussian matrix. However, our methods do not resort to a probabilistic analysis and are applicable for any given deterministic sensing matrix. Also, our algorithms have the strength of providing better bounds than existing methods [60, 110, 120, 169] for a wide range of matrix sizes.

2.1.1 Notations and preliminaries

We denote the sets of real numbers, and positive integers as $\mathbb{R}$ and $\mathbb{Z}^+$ respectively. We reserve uppercase letters K and L for index sets, and lowercase letters $k, l \in \mathbb{Z}^+$ for their respective cardinalities. We also use $|\cdot|$ to denote the cardinality of a set. We assume $k > l \geq 1$ throughout the paper. For vectors or scalars, we use lowercase letters, e.g., x, k, l. For a vector $x \in \mathbb{R}^n$, we use x_i for its i-th element. If we use an index set as a subscript of a vector, it represents the partial vector over the index set. For example, when $x \in \mathbb{R}^n$ and $K = \{1, 2\}$, x_K represents $[x_1, x_2]^T$. We reserve uppercase A for a sensing matrix whose dimension is $m \times n$. Since the number of columns of a sensing matrix A is n, the full index set we consider is $\{1, 2, ..., n\}$. In addition, we represent $\binom{n}{l}$ numbers of subsets as L_i, $i = 1, ..., \binom{n}{l}$, where $L_i \subset \{1, 2, ..., n\}$, $|L_i| = l$. We use the superscript * to represent an optimal solution of an optimization problem. For instance, z^* and K^* are the optimal solution of (2.5). Since we need to represent an optimal solution for each index set L_i, we use the superscript $i*$ to

represent an optimal solution for an index set L_i, e.g., z^{i*}. The maximum value of k such that both $\alpha_k < \frac{1}{2}$ and $\alpha_{k+1} \geq \frac{1}{2}$ hold true is denoted by the *maximum recoverable sparsity* k_{max}.

2.2 Pick-l-element Algorithm

Consider a sensing matrix with n columns. Then, there are $\binom{n}{k}$ subsets K each of cardinality k. When n and k are large, exhaustive search over these subsets to compute α_k is extremely expensive. For example, when $n = 100$ and $k = 10$, it takes a search over 1.7310e+13 subsets to compute α_k - a combinatorial task that is beyond the technological reach of common desktop computers. Our goal is to devise algorithms that can rapidly yield the exact value of α_k. As an initial step, we develop a method to compute an upper bound of α_k in polynomial time, which is called the *pick-l-element algorithm* (or simply, pick-l algorithm), where l is a chosen integer such that $1 \leq l < k$.

Let us define the proportion parameter for a given index set L such that $|L| = l$, denoted by $\alpha_{l,L}$, as

$$\alpha_{l,L} \triangleq \underset{\{z:\, Az=0,\, z\neq 0\}}{\text{maximize}} \frac{\|z_L\|_1}{\|z\|_1}. \tag{2.6}$$

(2.6) is the partial optimization problem of (2.4) only considering the vector z in the null space of A for a fixed index set L. We can obtain $\alpha_{l,L}$ by solving the following optimization problem:

$$\begin{aligned} &\underset{z}{\text{maximize}} \ \|z_L\|_1 \\ &\text{subject to } \|z\|_1 \leq 1,\ Az = 0. \end{aligned} \tag{2.7}$$

Since (2.7) is maximizing a convex function for a given subset L, we cast (2.7) as 2^l linear programming problems by considering all the possible sign patterns of every element of z_L (e.g., if $l = 2$ and $L = \{1, 2\}$, then, $\|z_L\|_1 = |z_1| + |z_2|$ can correspond to $2^l = 4$ possibilities: $z_1 + z_2$, $z_1 - z_2$, $-z_1 + z_2$, and $-z_1 - z_2$). $\alpha_{l,L}$ is equal to the maximum among the 2^l objective values.

The pick-l algorithm uses $\alpha_{l,L}$'s obtained from different index sets to compute an upper bound of α_k. Algorithm 2.1 shows the steps of the pick-l algorithm in detail. The following Lemmata show that the pick-l algorithm provides an upper bound of α_k. Firstly, we provide Lemma 2.1 to derive the upper bound of the proportion parameter for a fixed index set K, and then, we show that the pick-l algorithm yields an upper bound of α_k in Lemma 2.2.

Lemma 2.1 (Cheap Upper Bound (CUB) for a given subset K)**.** *Given a subset K, we have*

$$CUB(\alpha_{k,K}) \triangleq \frac{1}{\binom{k-1}{l-1}} \sum_{\{L_i \subseteq K,\, |L_i| = l\}} \alpha_{l,L_i} \geq \alpha_{k,K}. \tag{2.8}$$

Proof. Suppose that when $z = z^{i*}$ and $z = z^*$, we achieve the optimal value of (2.6) for given index sets L_i and K respectively, i.e., $\alpha_{l,L_i} = \frac{\|z^{i*}_{L_i}\|_1}{\|z^{i*}\|_1}$ and $\alpha_{k,K} = \frac{\|z^*_K\|_1}{\|z^*\|_1}$. Since each element of K appears $\binom{k-1}{l-1}$ times in $\{L_i \subseteq K,\, |L_i| = l\}$, we obtain the following inequality:

$$\alpha_{k,K} = \frac{\|z^*_K\|_1}{\|z^*\|_1} = \frac{1}{\binom{k-1}{l-1}} \sum_{\{L_i \subseteq K,\, |L_i| = l\}} \frac{\|z^*_{L_i}\|_1}{\|z^*\|_1} \leq \frac{1}{\binom{k-1}{l-1}} \sum_{\{L_i \subseteq K,\, |L_i| = l\}} \frac{\|z^{i*}_{L_i}\|_1}{\|z^{i*}\|_1} = CUB(\alpha_{k,K}).$$

The inequality is from the optimal value of (2.6) for each index set L_i. □

Algorithm 2.1 Pick-l-element algorithm, $1 \le l < k$ for computing an upper bound of α_k

1: Given a matrix A, calculate $\alpha_{l,L}$'s for all the subsets L, $|L| = l$, via (2.7).

2: Sort these $\binom{n}{l}$ different values of $\alpha_{l,L}$'s in descending order like (2.10).

3: Compute an upper bound of α_k via (2.9).

4: If the upper bound is less than $\frac{1}{2}$, then NSC for $k \in \mathbb{Z}^+$ is satisfied.

Lemma 2.2. *The pick-l algorithm provides an upper bound of α_k, namely*

$$\alpha_k \le \frac{1}{\binom{k-1}{l-1}} \sum_{i=1}^{\binom{k}{l}} \alpha_{l,L_i}, \tag{2.9}$$

$$\textit{where } \alpha_{l,L_1} \ge \alpha_{l,L_2} \ge \cdots \ge \alpha_{l,L_i} \ge \cdots \ge \alpha_{l,L_{\binom{n}{l}}}. \tag{2.10}$$

Proof. Without loss of generality, we assume that when $z = z^{i*}$, $i = 1, 2, ..., \binom{n}{l}$, α_{l,L_i}'s are obtained in descending order like (2.10). It is noteworthy that $\alpha_{k,K}$ is defined for a fixed K set; however, α_k is the maximum value over all the subsets with cardinality k. Suppose that when $z = z^*$ and $K = K^*$, α_k is achieved in (2.4). From the aforementioned definitions and similar argument as in Lemma 2.1, we have:

$$\alpha_k = \alpha_{k,K^*} \le \frac{1}{\binom{k-1}{l-1}} \sum_{\{L_i \subseteq K^*,\, |L_i| = l\}} \alpha_{l,L_i} \le \frac{1}{\binom{k-1}{l-1}} \sum_{i=1}^{\binom{k}{l}} \alpha_{l,L_i}.$$

The first inequality is from Lemma 2.1, and the last inequality is from the assumption that α_{l,L_i}'s are sorted in descending order. □

The steps 2 and 3 in Algorithm 2.1, which are sorting $\alpha_{l,L}$'s and computing the upper bound of α_k with sorted $\alpha_{l,L}$'s via (2.9), can also be done by solving the

following optimization problem without sorting operation:

$$\begin{aligned} &\underset{\gamma_i,\, 1\le i\le \binom{n}{l}}{\text{maximize}} \quad \sum_{i=1}^{\binom{n}{l}} \gamma_i \, \alpha_{l,L_i} \\ &\text{subject to} \quad 0 \le \gamma_i \le \frac{1}{\binom{k-1}{l-1}},\ 1 \le i \le \binom{n}{l}, \\ &\qquad\qquad \sum_{i=1}^{\binom{n}{l}} \gamma_i \le \frac{k}{l}. \end{aligned} \tag{2.11}$$

Here, we note that $\frac{1}{\binom{k-1}{l-1}} \times \binom{k}{l} = \frac{k}{l}$. Therefore, for the optimal value, the first $\binom{k}{l}$ largest α_{l,L_i}'s are chosen with the coefficient $\frac{1}{\binom{k-1}{l-1}}$.

The upshot of the pick-l algorithm is that we can reduce number of operations from $\binom{n}{k}$ enumerations to $\binom{n}{l}$. For example, when $n = 300$, $k = 20$, and $l = 2$, the number of operations is reduced by around 10^{26} times. Moreover, as n increases, the reduction rate increases. With the reduced enumerations, we can still have meaningful upper bounds of α_k through the pick-l-element algorithm. We will present the performance of the pick-l algorithm in Section 2.5 showing that the pick-l algorithm provides better upper bounds than the previous research [60, 110] even when $l = 2$. Furthermore, thanks to the pick-l algorithm, we can design a new algorithm based on a branch-and-bound search to calculate the exact α_k by using upper bounds of α_k obtained from the pick-l algorithm. It is noteworthy that the cheap upper bound introduced in Lemma 2.1 can provide upper bounds on $\alpha_{k,K}$ for specific subsets K, which enable our branch-and-bound method to calculate the exact α_k or more precise bounds on α_k. However, LP relaxation method [110] and SDP method [60] provide the upper bounds on α_k, instead of upper bounds on $\alpha_{k,K}$ for specific subsets K.

Finally, since we are also interested in k_{max}, we introduce the following Lemma

2.3 to bound the maximum recoverable sparsity k_{max}.

Lemma 2.3. *The maximum recoverable sparsity k_{max} satisfies*

$$k(\alpha_l) \triangleq \left\lceil l \cdot \frac{1/2}{\alpha_l} \right\rceil - 1 \leq k_{max}, \tag{2.12}$$

where $\lceil . \rceil$ is the ceiling function.

Proof. To prove this lemma, we will show that when $k = \left\lceil l \cdot \frac{1/2}{\alpha_l} \right\rceil - 1$, $\alpha_k < \frac{1}{2}$. This can be concluded from an upper bound of α_k given as follows:

$$\alpha_k = \alpha_{k,K^*} \leq \frac{1}{\binom{k-1}{l-1}} \sum_{\{L_i \subseteq K^*, |L_i|=l\}} \alpha_{l,L_i} \leq \frac{\binom{k}{l}}{\binom{k-1}{l-1}} \alpha_l = \alpha_l \cdot \frac{k}{l}. \tag{2.13}$$

Note that there are $\binom{k}{l}$ terms in the summation. From (2.13), if $\alpha_l \cdot \frac{k}{l} < \frac{1}{2}$, then $\alpha_k < \frac{1}{2}$. In other words, if $k < l \cdot \frac{1/2}{\alpha_l}$, then $\alpha_k < \frac{1}{2}$. Since k is a positive integer, when $k = \left\lceil l \cdot \frac{1/2}{\alpha_l} \right\rceil - 1$, $\alpha_k < \frac{1}{2}$. Therefore, the maximum recoverable sparsity k_{max} should be larger than or at least equal to $\left\lceil l \cdot \frac{1/2}{\alpha_l} \right\rceil - 1$. □

It is noteworthy that in [110, Section 4.2.B], the authors introduced lower bound on k based on α_1, i.e., $k(\alpha_1)$. However, in Lemma 2.3, we provide a more general result. Furthermore, in Lemma 2.3, instead of using α_l, we can use an upper bound of α_l to obtain the recoverable sparsity k; namely, $k(UB(\alpha_l)) = \left\lceil l \cdot \frac{1/2}{UB(\alpha_l)} \right\rceil - 1 \leq k_{max}$, where $UB(\alpha_l)$ represents an upper bound of α_l. Since the proof follows the same track as the proof of Lemma 2.3, we omit the proof.

2.3 Optimized Pick-l Algorithm

We can tighten the upper bound of α_k in the pick-l algorithm by replacing the constant factor $\frac{1}{\binom{k-1}{l-1}}$ in (2.9) with optimized coefficients at the cost of additional

complexity, which we call as the *optimized* pick-l algorithm. This optimized pick-l algorithm is mostly useful from a theoretical perspective. In practice, it gives improved but similar performance in calculating the upper bound of α_k to the basic pick-l algorithm described in Section 2.2. As a theoretical merit of the *optimized* pick-l algorithm, we can show that as l increases, the upper bound of α_k becomes smaller or stays the same.

The optimized pick-l algorithm provides an upper bound of α_k via the following optimization problem:

$$\begin{aligned}
&\underset{\gamma_i,\, 1\le i\le \binom{n}{l}}{\text{maximize}} \quad \sum_{i=1}^{\binom{n}{l}} \gamma_i\, \alpha_{l,L_i} \\
&\text{subject to} \quad \gamma_i \ge 0,\ 1 \le i \le \tbinom{n}{l}, \\
&\qquad\qquad \sum_{i=1}^{\binom{n}{l}} \gamma_i \le \frac{k}{l}, \\
&\qquad\qquad \sum_{\{i:\, B\subseteq L_i,\, 1\le i\le \binom{n}{l}\}} \gamma_i \le \frac{\binom{k-b}{l-b}}{\binom{k-1}{l-1}}, \quad \begin{array}{l}\forall\ b\in\mathbb{Z}^+ \text{ s.t. } 1\le b\le l,\\ \forall\ B \text{ with } |B|=b\end{array}.
\end{aligned} \tag{2.14}$$

The optimal value of (2.14) is an upper bound of α_k.

In the following lemmata, we show that the optimized pick-l algorithm produces an upper bound of α_k and this bound is tighter than the upper bound obtained from the basic pick-l algorithm. The last lemma establishes that as l increases, the upper bound of α_k decreases or stays the same.

Lemma 2.4. *The optimized pick-l algorithm provides an upper bound of α_k.*

Proof. The strategy to prove Lemma 2.4 is to show that one feasible solution of (2.14) gives an upper bound of α_k. Suppose when $K = K^*$, α_k is achieved, i.e., $\alpha_k = \alpha_{k,K^*}$.

For a feasible solution, let us choose $\gamma_i = \frac{1}{\binom{k-1}{l-1}}$ when $L_i \subseteq K^*$, and $\gamma_i = 0$ otherwise, which we can easily check whether it satisfies the first and second constraints of (2.14). For the third constraint, let us check the case when $b = l$ first. For $b = l$, we can choose an arbitrary index set B such that $|B| = b = l$. For the chosen B, there is only one L_i such that $B \subseteq L_i$, which is itself, i.e., $L_i = B$. For other chosen B's, it is the same. Hence, the third constraint represents

$$\gamma_i \leq \frac{1}{\binom{k-1}{l-1}},\ i = 1, 2, ..., \binom{n}{l}. \tag{2.15}$$

For $b = 1$, the third constraint represents

$$\sum_{\{i:\, B \subseteq L_i,\ 1 \leq i \leq \binom{n}{l},\ |B|=1\}} \gamma_i \leq 1. \tag{2.16}$$

Note that there are $\binom{n-1}{l-1}$ numbers of L_i's which have an index set B as a subset. Among $\binom{n-1}{l-1}$ numbers of γ_i's, only γ_i's whose corresponding L_i's are the subsets of K^* are $\frac{1}{\binom{k-1}{l-1}}$. Since each element in L_i such that $L_i \subseteq K^*$ appears $\binom{k-1}{l-1}$ times in $\{i :\ L_i \subseteq K^*,\ 1 \leq i \leq \binom{n}{l}\}$, the summation of γ_i, where the corresponding L_i's are the subset of K^*, becomes $\frac{1}{\binom{k-1}{l-1}} \times \binom{k-1}{l-1} = 1$, which satisfies (2.16). Basically, the third constraint makes that for an index, the summation of coefficients related to the index is limited to 1. In the same way, for $1 < b < l$, the chosen γ_i is a feasible solution of (2.14). From this feasible solution, we have $\frac{1}{\binom{k-1}{l-1}} \sum_{\{i:\, L_i \subseteq K^*,\, |L_i|=l\}} \alpha_{l,L_i}$ for the optimal value, which is an upper bound of α_k as shown in (2.13). □

Lemma 2.5. *The optimized pick-l algorithm provides a tighter, or at least the same, upper bound of α_k than the basic pick-l algorithm.*

Proof. We will show that the optimization problem (2.11) is a relaxation of (2.14). As in the proof of Lemma 2.4, for $b = l$, the third constraint of (2.14) represents (2.15), which is involved in the first constraint of (2.11). Since the third constraint of (2.14) considers other b values such that $1 \le b < l$, (2.14) has more constraints than (2.11). Therefore, the optimized pick-l algorithm, which is (2.14), provides a tighter or at least the same upper bound than the basic pick-l algorithm, which is (2.11). □

Lemma 2.6. *The optimized pick-l algorithm provides a tighter or at least the same upper bound than the optimized pick-p algorithm when $l > p$.*

Proof. We can upper bound the objective function of (2.14) by using (2.8) as follows:

$$\begin{aligned}
&\underset{\gamma_i,\, 1\le i\le\binom{n}{l}}{\text{maximize}} \ \frac{1}{\binom{l-1}{p-1}} \sum_{i=1}^{\binom{n}{l}} \gamma_i \sum_{\{j:\, P_j\subset L_i,\, |P_j|=p\}} \alpha_{p,P_j} \\
&\text{subject to } \ \gamma_i \ge 0,\ 1 \le i \le \tbinom{n}{l}, \\
&\qquad \sum_{i=1}^{\binom{n}{l}} \gamma_i \le \frac{k}{l}, \\
&\qquad \sum_{\{i:\, B\subseteq L_i,\, 1\le i\le\binom{n}{l}\}} \gamma_i \le \frac{\binom{k-b}{l-b}}{\binom{k-1}{l-1}}, \quad \begin{array}{l}\forall\ b\in\mathbb{Z}^+ \text{ s.t. } 1\le b\le l,\\ \forall\ B \text{ with } |B|=b\end{array}.
\end{aligned} \tag{2.17}$$

Note that in the objective function of (2.17), each $\alpha_{p,P_j},\ 1 \le j \le \binom{n}{p}$, appears $\binom{n-p}{l-p}$ times. Let us define

$$\gamma_j' \triangleq \frac{1}{\binom{l-1}{p-1}} \sum_{\{i:\, P_j\subset L_i,\, 1\le i\le\binom{n}{l}\}} \gamma_i.$$

We can relax (2.17) to the following problem, which turns out to be the same as the

optimized pick-p algorithm:

$$\begin{aligned}
&\underset{\gamma_j^{'},\, 1\le j\le\binom{n}{p}}{\text{maximize}} \ \sum_{j=1}^{\binom{n}{p}} \gamma_j^{'}\, \alpha_{p,P_j} \\
&\text{subject to } \ \gamma_j^{'} \ge 0,\ 1 \le j \le \tbinom{n}{p}, \\
&\qquad \sum_{j=1}^{\binom{n}{p}} \gamma_j^{'} \le \frac{k}{p}, \\
&\qquad \sum_{\{j:\, B\subseteq P_j,\, 1\le j\le\binom{n}{p}\}} \gamma_j^{'} \le \frac{\binom{k-b}{p-b}}{\binom{k-1}{p-1}}, \quad \begin{matrix}\forall\ b\in\mathbb{Z}^+ \text{ s.t. } 1\le b\le p,\\ \forall\ B \text{ with } |B|=b\end{matrix}.
\end{aligned} \tag{2.18}$$

The relaxation is shown by checking the constraints. The first constraint of (2.18) is trivial to obtain. For the second constraint, we can obtain the second constraint of (2.18) from the following relations:

$$\sum_{j=1}^{\binom{n}{p}} \gamma_j^{'} = \sum_{j=1}^{\binom{n}{p}} \frac{1}{\binom{l-1}{p-1}} \sum_{\{i:\, P_j\subset L_i,\, 1\le i\le\binom{n}{l}\}} \gamma_i = \frac{1}{\binom{l-1}{p-1}}\binom{l}{p}\sum_{i=1}^{\binom{n}{l}} \gamma_i \le \frac{1}{\binom{l-1}{p-1}}\binom{l}{p}\frac{k}{l} = \frac{k}{p},$$

where the second equality is obtained from the fact that γ_i, which is a coefficient of α_{l,L_i}, appears $\binom{l}{p}$ times in $\sum_{j=1}^{\binom{n}{p}} \sum_{\{i:\, P_j\subset L_i\}} \gamma_i$. The final inequality is from the second constraint of (2.17). The third constraint in (2.18) can be deduced from the following inequality:

$$\begin{aligned}
&\sum_{\{j:\, B\subseteq P_j,\, 1\le j\le\binom{n}{p}\}} \gamma_j^{'} \\
&= \frac{1}{\binom{l-1}{p-1}} \sum_{\{j:\, B\subseteq P_j,\, 1\le j\le\binom{n}{p}\}} \ \sum_{\{i:\, P_j\subset L_i,\, 1\le i\le\binom{n}{l}\}} \gamma_i \\
&= \frac{1}{\binom{l-1}{p-1}} \frac{\binom{n-b}{p-b}\binom{n-p}{l-p}}{\binom{n-b}{l-b}} \sum_{\{i:\, B\subset L_i,\, 1\le i\le\binom{n}{l}\}} \gamma_i \\
&\le \frac{1}{\binom{l-1}{p-1}} \frac{\binom{n-b}{p-b}\binom{n-p}{l-p}}{\binom{n-b}{l-b}} \frac{\binom{k-b}{l-b}}{\binom{k-1}{l-1}},\ 1\le b\le p \\
&= \frac{\binom{k-b}{p-b}}{\binom{k-1}{p-1}},\ 1\le b\le p,
\end{aligned}$$

where the second equality is from the fact that for a fixed P_j, there are $\binom{n-p}{l-p}$ numbers of L_i's, where $P_j \subset L_i$, $i = 1, ..., \binom{n}{l}$; for a fixed B, there are $\binom{n-b}{p-b}$ numbers of P_j's, where $B \subset P_j$, $j = 1, ..., \binom{n}{p}$, and $\binom{n-b}{l-b}$ numbers of L_i's, where $B \subset L_i$, $i = 1, ..., \binom{n}{l}$. Since (2.18) is obtained from the relaxation of (2.17), the optimal value of (2.18) is larger or equal to the optimal value of (2.17). (2.18) is just the optimized pick-p algorithm. Thus, when $l > p$, the optimized pick-l algorithm provides a tighter or at least the same upper bound than the optimized pick-p algorithm. □

By using larger l in the pick-l algorithm, we can obtain a tighter upper bound of α_k. However, this is only an upper bound of α_k, not its exact value. There is, however, a need to find the exact α_k in some applications [54, 83, 191]. To this end, we propose a new branch-and-bound based approach termed as the tree search algorithm to find the exact α_k. Our design of the tree search algorithm is enabled by the pick-l algorithms introduced in Sections 2.2 and 2.3.

2.4 Tree Search Algorithm

The Tree Search Algorithm (TSA) relies on representing subsets (with cardinality no bigger than k) of $\{1, ..., n\}$ using a tree structure. To find the index set K^* which leads to the maximum $\alpha_{k,K}$ (among all possible index set K's), TSA applies a best-first branch-and-bound search over the tree structure [57]. In the tree structure, each tree node J represents a subset of $\{1, ..., n\}$ with cardinality no bigger than k. For a tree node J, we calculate an upper bound of $\alpha_{k,K}$, which is valid for any child node K (with cardinality k) of the node J. If this upper bound happens to be smaller

than a lower bound of α_k, TSA will not further explore tree node J and all its child nodes.

Since TSA does not necessarily have to calculate $\alpha_{k,K}$ for every k-subset K, TSA can have less computational complexity than the Exhaustive Search Method (ESM). We will demonstrate the computational complexity of TSA in a statistical way in the numerical experiment section later. For simplicity of presentations, we introduce 1-Step TSA, which is based on the pick-1 algorithm. With simple modifications, we can extend 1-Step TSA to l-Step TSA, which is based on the pick-l algorithm for $l \geq 2$.

2.4.1 Tree structure

A node J in the tree represents an index subset of $\{1, ..., n\}$ such that $|J| \leq k$. An edge in the tree represents a set of indices (for 1-Step TSA, this set has cardinality 1), which we call *edge subset*. The index set that a node J represents is the union of all edge subsets that one encounters while traversing from the root node to the node J. Thus, we have the following rule:

[R1] A parent node is a subset of each of its child node(s).

A node that has no child is referred to as a *leaf node*. We call the cardinality of the index set corresponding to J as J's *height*. A tree is considered *fully expanded* if every leaf node has cardinality k. In constructing the tree structure, we impose the following "legitimate order", which ensures that any child node has a bigger index than its parent node.

[**R2**] "Legitimate order" - Let P, C, and E denote the parent node, child node, and edge subset between P and C respectively. Then, any index in P must be smaller than any index in E.

Fig. 2.1 illustrates this rule in a fully-expanded tree when $k = 2$ and $n = 3$.

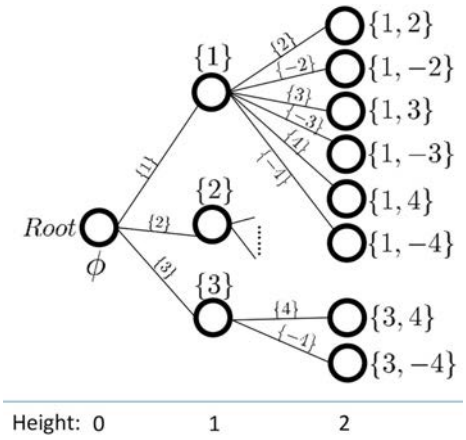


Figure 2.1: Fully expanded tree structure following the legitimate order for $k = 2$ and $n = 3$.

2.4.2 Branching node selection and tree expansion

For a tree node J, we calculate a value $B(J)$, which is no smaller than $\alpha_{k,K}$ for any child node K (with cardinality k) of the node J. $B(J)$ is calculated as

$$B(J) = \alpha_{j,J} + \sum_{i=1}^{t} \alpha_{1,\{i+max(J)\}}, \tag{2.19}$$

where $j + t = k$, $max(J)$ represents the largest index in J, and $\alpha_{1,\{1\}} \geq \alpha_{1,\{2\}} \geq ... \geq \alpha_{1,\{n\}}$. For example, in Fig. 2.1, for $k = 2$, $B(\{1\}) = \alpha_{1,\{1\}} + \alpha_{1,\{2\}}$. In each iteration, TSA selects a leaf node J with the largest $B(J)$ and branching a new node from the

node J or the parent of the node J. In order to further reduce the computational complexity of TSA, we introduce the following simple upper bound calculation instead of using (2.19) for the new node. For a new node Q attached to the node J in the tree, TSA computes $B(Q)$ as follows:

$$B(Q) = \alpha_{j,J} + \alpha_{1,Q \setminus J} + \sum_{i=1}^{t} \alpha_{1,\{i+max(Q)\}}, \tag{2.20}$$

where $j+t+1 = k$, $max(Q)$ represents the largest index in Q, and $\alpha_{1,\{1\}} \geq \alpha_{1,\{2\}} \geq ... \geq \alpha_{1,\{n\}}$. Thus, without calculating $\alpha_{j+1,Q}$, we can have a value $B(Q)$, which is still an upper bound of $\alpha_{k,K}$ for any child node K (with cardinality k) of the node Q.

Algorithm 2.2 shows detailed steps of TSA, based on the pick-1 algorithm (namely, 1-Step TSA). In the description, we define "expanding the tree from a node J" as follows:

[R3] "Expanding the tree from a node J" - Attaching a new node Q to the node J, where $B(Q)$ is the largest value defined as (2.20) among all the unattached child nodes of the node J.

2.4.3 Subtree pruning

The lower bound of α_k is calculated by simply computing an exact $\alpha_{k,K}$, which is $B(K)$ for a height-k leaf node K after calculating (2.19). Among the exact $\alpha_{k,K}$ values of height-k leaf nodes, TSA considers the biggest $\alpha_{k,K}$ value as the global lower bound of α_k. If a leaf node in any height has smaller $B(\cdot)$ value than the global lower bound, then, TSA prunes subtree of the leaf node and does not further explore all its child nodes.

2.4.4 Two enhancements

To further reduce the computational complexity of TSA, we incorporate two novel features to TSA. Firstly, when TSA attaches a new node Q to a node J in the tree structure, TSA computes $B(Q)$ as (2.20). Thus, without calculating $\alpha_{j+1,Q}$, we can still have a value $B(Q)$, which is still an upper bound of $\alpha_{k,K}$ for any child node K (with cardinality k) of the node Q.

Secondly, when TSA adds a new node Q to a node J in the tree structure (assuming $\alpha_{j,J}$ has already been calculated), TSA does not need to add all of J's child nodes to the tree at the same time. Instead, TSA only adds the node J's unattached child node Q with the largest $B(Q)$ as defined in (2.20). Namely, the index $Q \setminus J$ is no bigger than the index $Q' \setminus J$, where Q' is any unattached child of the node J. We note that $B(Q)$ is an upper bound on $B(Q')$ (according to (2.20)) for any other unattached child node Q' of the node J. Thus, $B(Q)$ is still an upper bound on $\alpha_{k,K}$ for any child node K of any of the node J's unattached child nodes. We note that the methods based on LP [110] and SDP [60] also provide upper bounds on α_k. However, they are unable to determine upper bounds of $\alpha_{k,K}$, which is for a specific index set K. This prevents use of LP and SDP methods in our branch-and-bound method for computing the exact α_k.

2.4.5 Advantage of tree search algorithm

Due to the nature of the branch and bound algorithm, we can obtain a global upper bound and a global lower bound of α_k while TSA runs. As the number of

Algorithm 2.2 Tree search algorithm based on the pick-1-element algorithm (1-Step TSA)

Input: $A \in \mathbb{R}^{m\times n}$, k, $l \leftarrow 1$ ▷ 1-Step TSA, i.e., $l = 1$
Output: α_k

▷ **Pre-commputation:**
compute $\alpha_{l,\{i\}}$ for $i = 1, ..., n$ via (2.7)
permute columns of A in descending order of $\alpha_{1,\{i\}}$'s, so that $\alpha_{1,\{1\}} \geq ... \geq \alpha_{1,\{n\}}$

▷ **Tree expansion:**
start with root node $\varnothing$, where $B(\varnothing) = \sum_{i=1}^{k} \alpha_{1,\{i\}}$, in a tree structure Υ
Loop
  $J \leftarrow$ a node that has the largest $B(\cdot)$ among all the leaf nodes in Υ
  $j \leftarrow |J|$
  if $\alpha_{j,J}$ *is not calculated* **then**
    compute $\alpha_{j,J}$ via (2.7) and update $B(J)$ via (2.19)
    expand Υ from the parent of J ▷ See **[R3]**
  else
    if $j = k$ **then**
      $\alpha_k \leftarrow B(J)$
      break
    else
      expand Υ from J ▷ See **[R3]**
    end
  end
EndLoop

iterations increases in TSA, we can obtain tighter and tighter upper bounds on α_k, which is the largest $B(\cdot)$ among leaf nodes. By using the global upper bound of α_k, we can obtain a lower bound of the recoverable sparsity k via Lemma 2.3. Thus, even though the complexity of TSA is too high to finish in a timely manner, we can still obtain new bounds on the recoverable sparsity k by early terminating TSA.

2.5 Numerical Experiments

We conducted extensive simulations to compute the exact α_k or its upper bounds using TSA and the pick-l algorithms. For the same matrices, we compared our

methods with the previously mentioned approaches that use LP relaxation [110] and SDP method [60]. We assess the computational complexity in terms of execution time of the algorithms.[1] In addition, we carried out numerical experiments to demonstrate the computational complexity of TSA statistically.

For LP method in [110] and SDP method in [60], we used the Matlab codes[2] provided by the authors. Since these methods use different solvers, we used two different solvers: CVX [91] - a package for specifying and solving convex programs - for SDP method and MOSEK [141] - a commercial LP solver - for LP method. For our methods, we used MOSEK to solve (2.7). Also, to be consistent with the previous research, matrices were generated from the Matlab code provided by the authors of [60] at `http://www.di.ens.fr/~aspremon/NSPcode.html`.

2.5.1 Performance comparison

Firstly, we considered Gaussian matrices and partial Fourier matrices sized from $n = 40$ to $n = 256$. We chose $n = 40$ so that our results can be compared with the simulation results in [60].

2.5.1.1 Low-dimensional sensing matrices

Sensing matrices with $n = 40$: We considered sensing matrices of row dimension $m = 0.5n$, $0.6n$, $0.7n$, $0.8n$, where $n = 40$. For every matrix size, 10 different

[1]We conducted our experiments on HP Z220 CMT with Intel Core i7-3770 dual core CPU @3.4GHz clock speed and 16GB DDR3 RAM, using Matlab (R2013b) on Windows 7.

[2]LP method from `http://www2.isye.gatech.edu/~nemirovs/` and SDP method from `http://www.di.ens.fr/~aspremon/NSPcode.html`.

Table 2.1: α_k comparison - Gaussian Matrix

(Floor to the nearest hundredth)

A $(m\times n)$	Algo.	α_1	α_2	α_3	α_4	α_5	k_{max}[c]
20×40	pick-1	0.28	0.54	0.80	1	1	1/1.1
	pick-2	0.28	0.45	0.65	0.84	1	2/1.9
	pick-3	0.28	0.45	0.57	0.75	0.91	2/1.9
	TSA	0.28	0.45	0.57	0.67	0.75	2/1.9
	LP[a]	0.28	0.49	0.66	0.83	0.97	2/1.6
	SDP[b]	0.28	0.48	0.65	0.80	0.94	2/1.8
24×40	pick-1	0.23	0.45	0.66	0.86	1	2/2.0
	pick-2	0.23	0.37	0.52	0.68	0.84	2/2.1
	pick-3	0.23	0.37	0.46	0.60	0.74	3/2.8
	TSA	0.23	0.37	0.46	0.57	0.65	3/2.8
	LP	0.23	0.40	0.55	0.70	0.83	2/2.0
	SDP	0.23	0.40	0.55	0.69	0.82	2/2.0
28×40	pick-1	0.18	0.35	0.52	0.69	0.85	2/2.0
	pick-2	0.18	0.31	0.45	0.58	0.71	3/3.0
	pick-3	0.18	0.31	0.41	0.53	0.65	3/3.0
	TSA	0.18	0.31	0.41	0.49	0.57	4/3.5
	LP	0.18	0.33	0.48	0.60	0.71	3/3.0
	SDP	0.18	0.33	0.47	0.59	0.70	3/3.0
32×40	pick-1	0.14	0.28	0.41	0.54	0.66	3/3.0
	pick-2	0.14	0.24	0.36	0.46	0.57	4/3.8
	pick-3	0.14	0.24	0.33	0.43	0.52	4/4.1
	TSA	0.14	0.24	0.33	0.40	0.47	5/4.9
	LP	0.14	0.26	0.37	0.48	0.57	4/3.9
	SDP	0.14	0.26	0.37	0.47	0.56	4/4.0

[a] Linear Programming [110] [b] Semidefinite Programming [60] [c] median / average

realizations each of Gaussian and partial Fourier matrices were considered. So, we used total 80 different n = 40 sensing matrices for the numerical experiments in Tables 2.1, 2.2, 2.3 and 2.4. We normalized all of the matrix columns so that they have a unit ℓ_2-norm. The entries of Gaussian matrices were i.i.d standard Gaussian $\mathcal{N}(0,1)$. The partial Fourier matrices had m rows as the partial Fourier bases drawn at random. We compared our algorithms - pick-1, pick-2, pick-3 and TSA - to LP and SDP methods.

Table 2.2: Execution time (geometric mean) - Gaussian Matrix

(Unit: second)

A $(m\times n)$	Algo.	α_1	α_2	α_3	α_4	α_5
20×40	pick-1	0.28	0.28	0.28	0.28	0.28
	pick-2	0.28	9.68	9.68	9.68	9.68
	pick-3	0.28	9.68	233.83	233.83	233.83
	TSA	1.20[a]	11.13[b]	11.38[b]	43.44[b]	551.18[c]
	LP	0.42	0.42	0.42	0.41	0.41
	SDP	46.72	74.90	64.27	59.05	56.33
	ESM	0.28	9.68	233.83	4.54e3[d]	6.31e4[d]
24×40	pick-1	0.28	0.28	0.28	0.28	0.28
	pick-2	0.28	9.85	9.85	9.85	9.85
	pick-3	0.28	9.85	248.95	248.95	248.95
	TSA	1.01[a]	10.99[b]	11.67[b]	23.37[b]	301.56[c]
	LP	0.46	0.46	0.46	0.44	0.46
	SDP	50.93	59.58	49.68	46.15	44.62
	ESM	0.28	9.85	248.95	4.82e3[d]	6.72e4[d]
28×40	pick-1	0.36	0.36	0.36	0.36	0.36
	pick-2	0.36	10.56	10.56	10.56	10.56
	pick-3	0.36	10.56	259.92	259.92	259.92
	TSA	1.18[a]	11.87[b]	11.96[b]	14.50[b]	290.96[c]
	LP	0.51	0.50	0.49	0.50	0.50
	SDP	48.89	48.94	42.32	39.00	38.83
	ESM	0.36	10.56	259.92	5.21e3[d]	7.13e4[d]
32×40	pick-1	0.36	0.36	0.36	0.36	0.36
	pick-2	0.36	11.35	11.35	11.35	11.35
	pick-3	0.36	11.35	281.42	281.42	281.42
	TSA	1.14[a]	13.92[b]	12.93[b]	14.01[b]	307.99[c]
	LP	0.54	0.53	0.54	0.54	0.54
	SDP	47.76	47.96	39.92	36.62	34.12
	ESM	0.36	11.35	281.42	5.60e3[d]	7.92e4[d]

[a] 1-Step TSA [b] 2-Step TSA [c] 3-Step TSA

[d] Exhaustive Search Method (Estimated time = avg. time to solve (2.7) × # of index sets)

Table 2.3: α_k comparison - Partial Fourier Matrix

(Floor to the nearest hundredth)

A $(m\times n)$	Algo.	α_1	α_2	α_3	α_4	α_5	k_{max} [c]
20×40	pick-1	0.19	0.39	0.59	0.78	0.98	2/2.0
	pick-2	0.19	0.34	0.51	0.68	0.85	2/2.3
	pick-3	0.19	0.34	0.47	0.63	0.78	3/2.8
	TSA	0.19	0.34	0.47	0.60	0.70	3/2.8
	LP[a]	0.19	0.39	0.59	0.78	0.98	2/2.0
	SDP[b]	0.19	0.39	0.59	0.78	0.98	2/2.0
24×40	pick-1	0.16	0.32	0.48	0.64	0.80	3/2.7
	pick-2	0.16	0.29	0.43	0.58	0.72	3/2.9
	pick-3	0.16	0.29	0.40	0.53	0.66	3/3.1
	TSA	0.16	0.29	0.40	0.49	0.60	4/3.4
	LP	0.16	0.32	0.48	0.64	0.80	3/2.7
	SDP	0.16	0.32	0.48	0.64	0.80	3/2.7
28×40	pick-1	0.12	0.24	0.37	0.49	0.61	4/3.7
	pick-2	0.12	0.23	0.34	0.46	0.57	4/3.9
	pick-3	0.12	0.23	0.32	0.43	0.54	4/3.9
	TSA	0.12	0.23	0.32	0.41	0.50	4/4.0
	LP	0.12	0.24	0.37	0.49	0.61	4/3.7
	SDP	0.12	0.24	0.37	0.49	0.61	4/3.7
32×40	pick-1	0.09	0.18	0.28	0.37	0.46	5/4.7
	pick-2	0.09	0.17	0.26	0.35	0.44	5/4.7
	pick-3	0.09	0.17	0.24	0.33	0.41	5/4.7
	TSA	0.09	0.17	0.24	0.32	0.38	5/4.7
	LP	0.09	0.18	0.28	0.37	0.46	5/4.7
	SDP	0.09	0.18	0.28	0.37	0.46	5/4.7

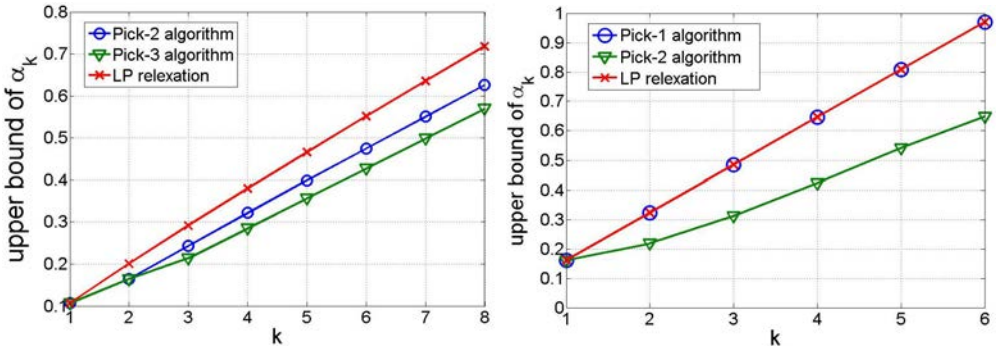
[a] Linear Programming [110] [b] Semidefinite Programming [60] [c] median / average

For a matrix of given size and type, k was increased from 1 to 5 in unit steps. Tables 2.1 and 2.3 show the *median* value of α_k and obtained k_{max} on Gaussian matrices and partial Fourier matrices respectively. For the recoverable sparsity k_{max}, we provide the median value obtained by using the median value of α_k and average value calculated from ten k_{max} from ten random cases. To be consistent with the previous research [60], in which the authors used the median value of α_k to compare

SDP method to LP method, we used the median value obtained over 10 random realizations of sensing matrix and derived the largest recoverable sparsity from the median value of α_k. Compared to the previous works involving LP and SDP methods, we obtained bigger or at least the same recoverable sparsity k_{max} by using pick-2, the pick-3 and TSA. Also, we obtained exact α_k values for $k = 1, 2, ..., 5$ by using TSA, while LP and SDP methods only provided an exact α_k for $k = 1$.

For execution time, we calculated the *geometric mean* to avoid biases for the average. Tables 2.2 and 2.4 list the averaged execution time. Given our computational resources, we also estimated execution time to find the exact α_k using ESM. For the estimated execution time, we obtained average time to solve (2.7) from 100 trials and multiplied the average time to total number of index sets. For example, ESM takes around 21 hours ($= 0.1203$ sec. $\times \binom{40}{5}$) to find α_5 for one instance of 32×40 Gaussian matrix, where 0.1203 sec. was obtained by running 100 random cases and calculating the arithmetic mean. 3-Step TSA reduced the computational time in average around 250 times in order to obtain the exact α_5.

Sensing matrices with $n = 256$: We assessed the performance of the pick-l algorithm for sensing matrices with $n = 256$. We carried out numerical experiments on 128×256 Gaussian matrices in Fig. 2.2 (a) and 64×256 partial Fourier matrices in Fig. 2.2 (b). Here, for 10 sensing matrices, we obtained the median values of the upper bounds of α_k using the pick-l algorithms and compared the results with the LP relaxation method [110]. We omitted the SDP method [60] from this experiment due to its very high computational complexity for $n = 256$. For the pick-3 algorithm



(a) 128×256 Gaussian matrices (b) 64×256 Partial Fourier matrices

Figure 2.2: Median upper bounds of α_k from the pick-l algorithm and the LP relaxation method.

in Fig. 2.2 (a), we calculated the upper bound of α_3 via TSA, and used this result to calculate upper bounds of α_k, $k = 3, 4, ..., 8$ via (2.13). Fig. 2.2 (a) and (b) demonstrate that, with an appropriate choice of l, the upper bound of α_k obtained via the pick-l algorithm can be tighter than that from the LP relaxation method. For example, for 128×256 Gaussian matrices, LP relaxation often determines the maximum recoverable sparsity as 5, while the pick-2 algorithm improves it to 6. In the pick-3 algorithm, the maximum recoverable sparsity is 7 ($\alpha_7 = 0.49$). For 64×256 partial Fourier matrices, the maximum recoverable sparsity from LP relaxation and the pick-2 algorithm are 3 and 4 respectively.

2.5.1.2 High-dimensional sensing matrices

Sensing matrix with $n \geq 1024$: We conducted numerical experiments for Gaussian sensing matrices from $n = 1024$ to $n = 6144$. For these numerical experiments in Table 2.5, we calculated the lower bound on the recoverable sparsity k

and obtained the corresponding execution time. SDP method [60] was not available to these experiments due to its very high computational complexity. For an upper bound of α_2, denoted by $UB(\alpha_2)$, in Table 2.5, we ran TSA for 1 day (24 hours) and obtained the upper bound of α_2. With the upper bound of α_2, we obtained the lower bound of k via Lemma 2.3.

Our numerical results in Tables 2.5 and 2.6 clearly show that our pick-l algorithm outperforms LP and SDP based methods in recoverable sparsity k and execution time. We note that although our pick-1 algorithm provides the same recoverable sparsity k as LP method [110] in Tables 2.5 and 2.6, the complexity of LP method is higher than our method on $m \times n$ Gaussian matrices, where $m > 1/2n$. We achieved this result by using the null space of the sensing matrix A instead of directly using the sensing matrix A [52].

Even for extremely large sensing matrices, e.g, 4014×4096 and 6021×6144, where LP and SDP cannot provide any lower bound on k due to unreasonable computational time, our pick-l algorithm can provide the lower bound on k efficiently. Table 2.6 shows the lower bound on k and the execution time in detail for these large dimensional matrices, where our verified recoverable sparsity k can be as large as 558 on 6134×6144 sensing matrix. We obtained the estimated time for LP method by running Matlab code that the authors provided, which provides the percentage of the calculation.

2.5.2 Complexity of tree search algorithm

In this subsection, we carried out numerical experiments to demonstrate the computational complexity of the tree search algorithm statistically on randomly chosen Gaussian sensing matrices. Fig. 2.3 (a) and (b) show simulation results about the distribution of execution time and number of nodes located in height 5 in TSA respectively. For $m = 0.5n$, we generated 100 random realizations of Gaussian matrices and computed α_5 using 3-Step TSA. The maximum number of leaf node whose cardinality is k is $\binom{n}{k} = \binom{40}{5} = 6.58008e5$. From Fig. 2.3 (b), we note that for 90 % of the cases, TSA terminates before 1.6 % of height-5 nodes are attached to the tree.

In addition, we provide the execution time of TSA for different n with fixed k in Fig. 2.4, and for different k with fixed n in Fig. 2.5 on randomly chosen Gaussian matrices. In order to obtain the exact α_k, we achieved around 100 times faster execution time via TSA than ESM for $k = 3, 4$.

Fig. 2.6 shows an illustration of the global lower bound and global upper bound in TSA on 20×40 Gaussian sensing matrix.

2.5.3 Application to network tomography problem

We apply our new tools introduced in this paper to verify NSC for sensing matrices in network tomography problems [36, 54, 83, 175, 179, 191]. In an undirected complete graph model for the communication network, the communication delay over each link can be determined by sending packets through probing paths that are composed of connected links. The delay of each path is then measured by adding the

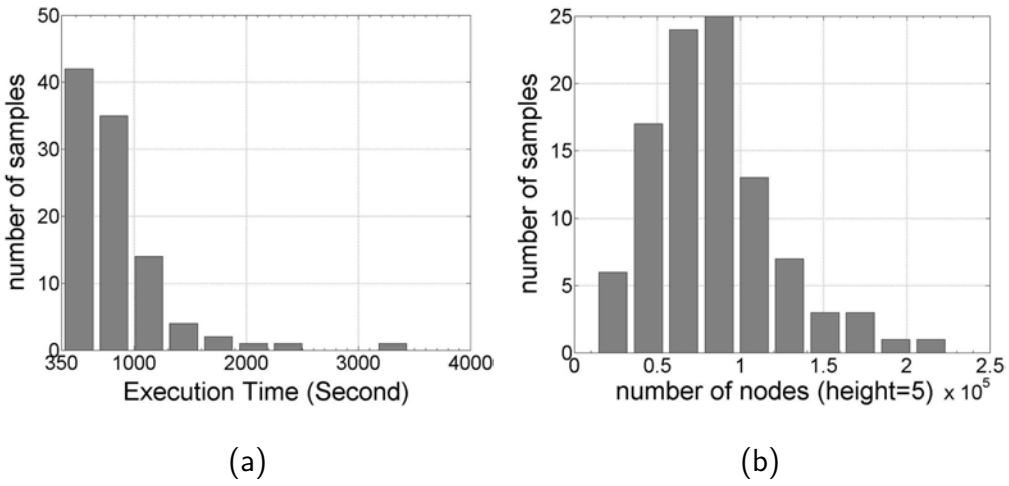
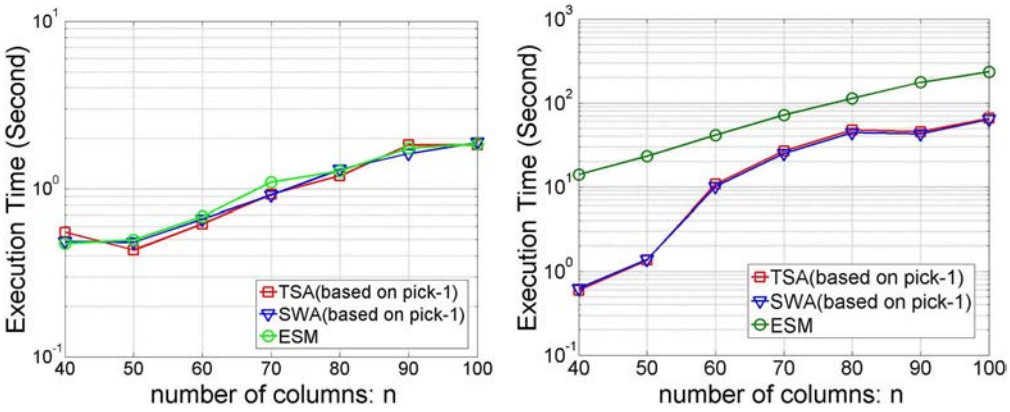


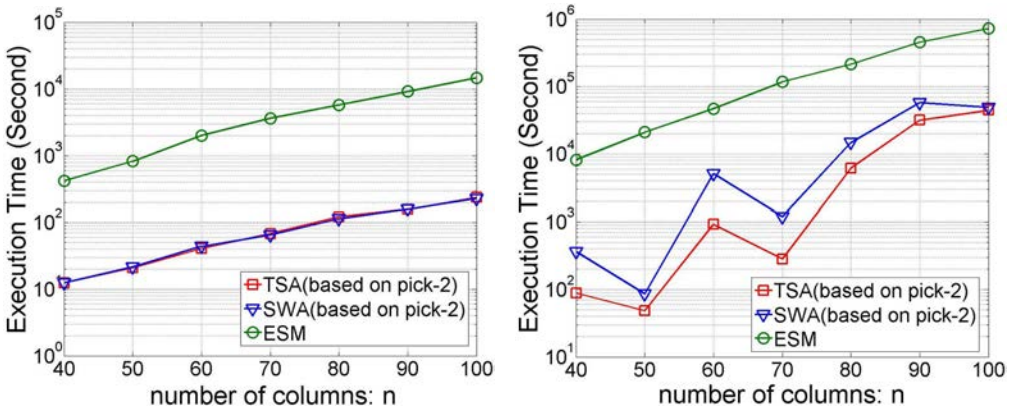
Figure 2.3: Histograms of TSA (based on the pick-3) to find α_5 on 100 randomly chosen 20×40 Gaussian sensing matrices for each method. (a) Execution time of TSA. (b) Number of nodes in TSA.

delays over its links. Generally most links are uncongested, and only a few congested links have significant delays. It is, therefore, reasonable to think of finding the link delays as a sparse recovery problem. This sparse problem can be expressed in a system of linear equations $y = Ax$, where the vector $y \in \mathbb{R}^m$ is the delay of m paths, the vector $x \in \mathbb{R}^n$ is the delay vector for the n links, and A is a sensing matrix. The element A_{ij} of A is 1, if and only if path y_i, $i \in \{1, 2, ..., m\}$, goes through link j, $j \in \{1, 2, ..., n\}$; otherwise A_{ij} equal to 0 (see Fig. 2.7). The indices of nonzero elements in the vector x correspond to the congested links.

In our numerical experiments to verify NSC in network tomography problems, the paths for sending data packets were generated by random walks of fixed length. Table 2.7 summarizes the results of our experiments. We note that by using TSA, one can *exactly* verify that a total of $k = 2$ and $k = 4$ congested link delays can be



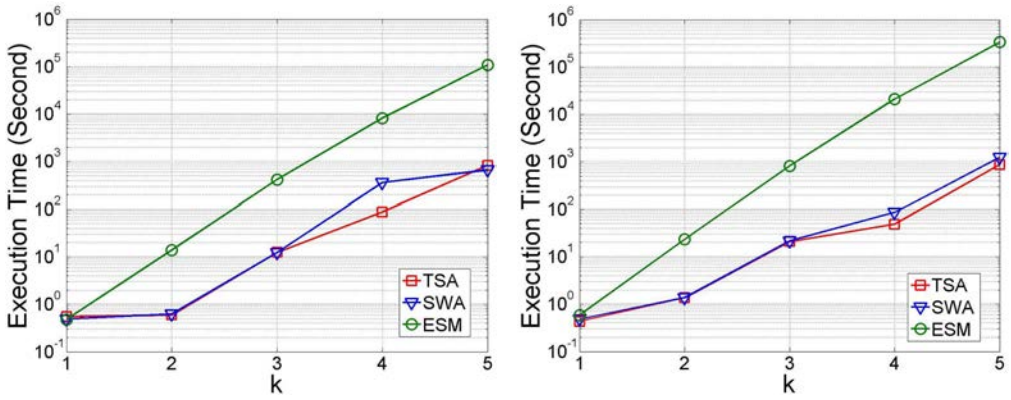
(a) $k = 1$ (b) $k = 2$



(c) $k = 3$ (d) $k = 4$

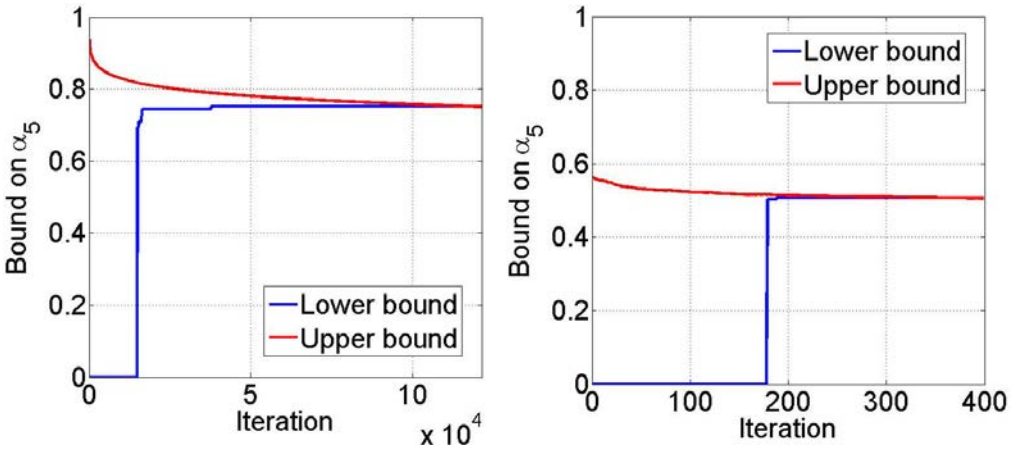
Figure 2.4: The execution time of TSA in log scale as a function of n on randomly chosen $m \times n$ Gaussian matrices, where $m = n/2$.

uniquely found by solving ℓ_1 minimization problem (2.2) for the randomly generated network measurement matrices 33×66 (12-node complete graph) and 53×105 (15-node complete graph) respectively. For a 53×105 matrix, 3-Step TSA substantially reduced the execution time to find α_5 around 200 times compared to ESM.



(a) $n = 40$ (b) $n = 50$

Figure 2.5: The execution time of TSA in log scale as a function of k on randomly chosen $m \times n$ Gaussian matrices, where $m = n/2$.



(a) 20×40 (b) 32×40

Figure 2.6: Global lower bound and global upper bound in TSA on Gaussian sensing matrix A for $(k, l) = (5, 3)$.

Table 2.4: Execution time (geometric mean) - Partial Fourier Matrix

(Unit: second)

A $(m\times n)$	Algo.	α_1	α_2	α_3	α_4	α_5
20×40	pick-1	0.25	0.25	0.25	0.25	0.25
	pick-2	0.25	8.97	8.97	8.97	8.97
	pick-3	0.25	8.97	229.37	229.37	229.37
	TSA	1.00[a]	10.28[b]	16.77[b]	107.73[b]	1769.29[c]
	LP	0.37	0.36	0.35	0.38	0.38
	SDP	29.70	71.82	70.81	67.26	61.16
	ESM	0.25	8.97	229.37	4.61e3[d]	6.56e4[d]
24×40	pick-1	0.27	0.27	0.27	0.27	0.27
	pick-2	0.27	9.57	9.57	9.57	9.57
	pick-3	0.27	9.57	243.33	243.33	243.33
	TSA	1.06[a]	10.75[b]	19.12[b]	72.58[b]	1042.96[c]
	LP	0.40	0.38	0.39	0.40	0.41
	SDP	29.24	52.10	53.47	59.83	57.07
	ESM	0.27	9.57	243.33	4.83e3[d]	6.84e4[d]
28×40	pick-1	0.34	0.34	0.34	0.34	0.34
	pick-2	0.34	9.86	9.86	9.86	9.86
	pick-3	0.34	9.86	247.52	247.52	247.52
	TSA	0.96[a]	10.89[b]	14.17[b]	27.60[b]	353.26[c]
	LP	0.44	0.44	0.46	0.44	0.44
	SDP	29.64	43.04	43.58	46.23	42.38
	ESM	0.34	9.86	247.52	5.09e3[d]	7.37e4[d]
32×40	pick-1	0.34	0.34	0.34	0.34	0.34
	pick-2	0.34	10.51	10.51	10.51	10.51
	pick-3	0.34	10.51	263.06	263.06	263.06
	TSA	1.12[a]	11.74[b]	16.16[b]	36.35[b]	437.79[c]
	LP	0.48	0.47	0.46	0.49	0.47
	SDP	27.29	31.69	36.33	31.67	31.24
	ESM	0.34	10.51	263.06	5.38e3[d]	7.63e4[d]

[a] 1-Step TSA [b] 2-Step TSA [c] 3-Step TSA

[d] Exhaustive Search Method (Estimated time = avg. time to solve (2.7) × # of index sets)

Table 2.5: Lower bound on k and execution time (Gaussian Matrix with $n = 1024$)

matrix A	Pick-1	$k(UB(\alpha_2)^{b})$	$k(\alpha_1)$	LP[a]
102×1024	2	3	2	2
205×1024	4	4	4	4
307×1024	5	6	5	5
410×1024	7	8	7	7
512×1024	9	10	9	9
614×1024	12	13	12	12
717×1024	16	17	15	16
819×1024	21	23	20	21
922×1024	32	36	30	32

(a) Lower bound on k

matrix A	Pick-1	$k(UB(\alpha_2)^{b})$	$k(\alpha_1)$	LP[a]
102×1024	237	1 day	237	200
205×1024	452	1 day	452	429
307×1024	796	1 day	796	723
410×1024	1207	1 day	1207	1073
512×1024	1952	1 day	1952	1600
614×1024	2150	1 day	2150	2217
717×1024	1337	1 day	1337	2992
819×1024	838	1 day	838	3904
922×1024	386	1 day	386	4730

(b) Execution time (Unit: second)

[a] Linear Programming [110] [b] Upper bound of α_2 obtained from 1-Step TSA after 1 day's run

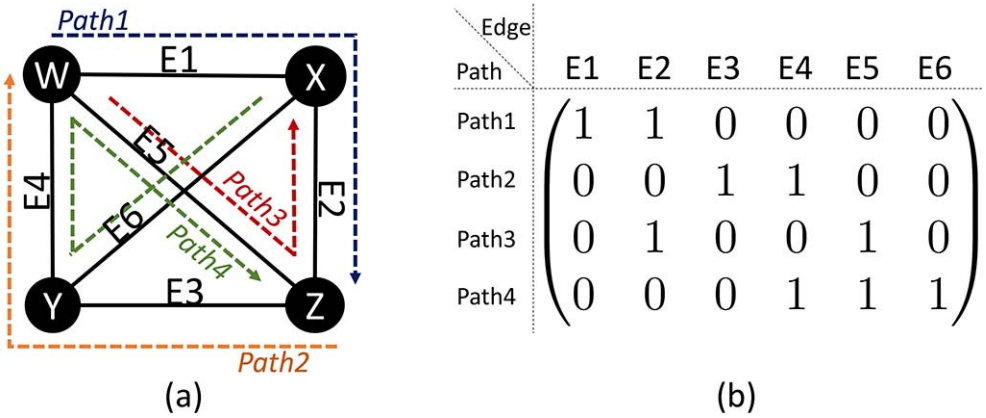


Figure 2.7: (a) A simple example of a network tomography graph. W, X, Y, and Z are nodes in the network, and $Path1$, 2, 3, and 4 are the probing paths through which the packets are sent. (b) The sensing matrix corresponding to the graph shown in (a). The rows and columns of the matrix represent probing paths and edges respectively.

Table 2.6: Lower bound on k and execution time (Gaussian Matrix)

matrix A	Pick-1	$k(\alpha_1)$	LP[a]
2007×2048	102	90	102
4014×4096	152	139	N/A[b]
6021×6144	190	174	N/A
6134×6144	558	406	N/A

(a) Lower bound on k

matrix A	Pick-1	$k(\alpha_1)$	LP
2007×2048	671	671	71948
4014×4096	9116	9116	15 days[c]
6021×6144	38935	38935	65.5 days[d]
6134×6144	13734	13734	41.7 days[e]

(b) Execution time (Unit: second)

[a] Linear Programming [110] [b] Not Available [c] Estimated time (15 hours for 4% calculations)

[d] Estimated time (15 hours for 1% calculations) [e] Estimated time (10 hours for 1% calculations)

Table 2.7: α_k and execution time in network tomography problems

matrix A	Algo.	α_1	α_2	α_3	α_4	α_5	k_{max}
33×66	TSA	0.28	0.41	0.50	0.57	0.62	2
53×105	TSA	0.20	0.29	0.36	0.45	0.52	4

(a) α_k values (Floor to the nearest hundredth)

matrix A	Algo.	α_1	α_2	α_3	α_4	α_5
33×66	TSA	1.4	3.7	33.1	457.3	3.0e4
	ESM	1.3	42.9	915.2	1.4e4	1.8e5
53×105	TSA	2.1	24.4	125.8	834.8	1.0e4
	ESM	2.1	109.2	3.7e3	9.6e4	1.9e6

(b) Execution time (Unit: second)

[a] 1-Step TSA [b] 2-Step TSA [c] 3-Step TSA

[d] Exhaustive search method (Estimated Operation time = avg. time to solve (2.7) (=0.02 sec.) for an index set × total number of index sets)

CHAPTER 3
SUPER-RESOLUTION FOR SPECTRALLY SPARSE SIGNALS

3.1 Spectral Super-resolution With Prior Knowledge

3.1.1 Introduction

In many areas of engineering, it is desired to infer the spectral contents of a measured signal. In the absence of any *a priori* knowledge of the underlying statistics or structure of the signal, the choice of spectral estimation technique is a subjective craft [134, 163]. However, in several applications, the knowledge of signal characteristics is available through previous measurements or prior research. By including such prior knowledge during spectrum estimation process, it is possible to enhance the performance of spectral analysis.

One useful signal attribute is its sparsity in spectral domain. In recent years, spectral estimation methods that harness the spectral sparsity of signals have attracted considerable interest [27, 71, 136, 168]. These methods trace their origins to *compressed sensing* (CS) that allows accurate recovery of signals sampled at sub-Nyquist rate [66]. In the particular context of spectral estimation, the signal is assumed to be sparse in a finite discrete dictionary such as Discrete Fourier Transform (DFT). As long as the true signal frequency lies in the center of a DFT bin, the discretization in frequency domain faithfully represents the continuous reality of the true measurement. If the true frequency is not located on this discrete frequency grid, then the aforementioned assumption of sparsity in the DFT domain is no longer

valid [101, 166]. The result is an approximation error in spectral estimation often referred to as scalloping loss [93], basis mismatch [49], and gridding error [79].

Recent state-of-the-art research [27, 167, 168] has addressed the problem of basis mismatch by proposing compressed sensing in continuous spectral domain. This *grid-free* approach is inspired by the problems of total variation minimization [27] and atomic norm minimization [168] to recover *super-resolution* frequencies - lying anywhere in the continuous domain $[0, 1]$ - with few random time samples of the spectrally sparse signal, provided the line spectrum maintains a nominal separation. A number of generalizations of off-the-grid compressed sensing for specific signal scenarios have also been attempted, including extension to higher dimensions [46, 48, 187, 194].

However, these formulations of off-the-grid compressed sensing assume no prior knowledge of signal other than sparsity in spectrum. In fact, in many applications, where signal frequencies lie in continuous domain such as radar [161], acoustics [173], communications [13], and power systems [202], additional prior information of signal spectrum might be available. For example, a radar engineer might know the characteristic speed with which a fighter aircraft flies. This knowledge then places the engineer in a position to point out the ballpark location of the echo from the aircraft in the Doppler frequency spectrum. Similarly, in a precipitation radar, the spectrum widths of echoes from certain weather phenomena (tornadoes or severe storms) are known from previous observations [70]. This raises the question whether we can use signal structures beyond sparsity to improve the performance of spectrum estimation.

There are extensive works in compressed sensing literature that discuss recovering sparse signals using secondary signal support structures, such as *structured sparsity* [38] (tree-sparsity [8], block sparsity [164], and Ising models [37]), spike trains [6, 95], nonuniform sparsity [114, 181], and multiple measurement vectors (MMVs) [72]. However, these approaches assume discrete-valued signal parameters while, in the spectrum estimation problem, frequencies are continuous-valued. Therefore, the techniques of using prior support information in discrete compressed sensing for structured sparsity do not directly extend to spectrum estimation. Moreover, it is rather unclear as to how general signal structure constraints can be imposed for super-resolution recovery of continuous-valued frequency components.

In this paper, we focus on a more generalized approach to super-resolution that addresses the foregoing problems with line spectrum estimation. We propose continuous-valued line spectrum estimation of irregularly undersampled signal in the presence of structured sparsity. Prior information about the signal spectrum comes in various forms. For example, in the spectral information concerning a rotating mechanical system, the frequencies of the supply lines or interfering harmonics might be precisely known [185]. However, in a communication problem, the engineer might only know the frequency band in which a signal frequency is expected to show up. Often the prior knowledge is not even specific to the level of knowing the frequency subbands precisely. The availability of previous measurements, such as in remote sensing or bio-medicine, can aid in knowing the likelihood of having an active signal frequency in the neighborhood of a specific spectral band. In this paper, we greatly

broaden the scope of prior information that can range from knowing only the likelihood of occurrence of frequency components in a spectral subband to exactly knowing the location of some of the frequencies.

In all these cases, we propose a precise semidefinite program to perfectly recover all the frequency components. When some frequencies are *precisely* known, we propose to use *conditional atomic norm* minimization to recover the off-the-grid frequencies. In practice, the frequencies are seldom *precisely* known. However, as long as the frequency locations are approximately known to the user, we show that the spectrally sparse signal could still be perfectly reconstructed. Here, we introduce *constrained atomic norm* minimization that accepts the *block priors* - frequency subbands in which true spectral contents of the signal are known to exist - in its semidefinite formulation. When only the probability density function of signal frequencies is known, we incorporate such a *probabilistic prior* in the spectral estimation problem by suggesting the minimization of *weighted atomic norm*. The key is to transform the dual of atomic norm minimization to a semidefinite program using linear matrix inequalities (LMI). These linear matrix inequalities are, in turn, provided by theories of positive trigonometric polynomials [81]. Our methods boost the signal recovery by admitting lesser number of samples for spectral estimation and decreasing reliance on the minimum resolution necessary for super-resolution. If the prior information locates the frequencies within very close boundaries of their true values, then we show that it is possible to perfectly recover the signal using samples no more than thrice the number of signal frequencies.

Our work has close connections with a rich heritage of research in spectral estimation. For uniformly sampled or *regularly spaced* signals, there are a number of existing approaches for spectral estimation by including known signal characteristics in the estimation process. The classical Prony's method can be easily modified to account for known frequencies [173]. Variants of the subspace-based frequency estimation methods such as MUSIC (MUltiple SIgnal Classification) and ESPRIT (Estimation of Signal Parameters via Rotation Invariance Techniques) have also been formulated [123, 185], where prior knowledge can be incorporated for parameter estimation. For applications wherein only approximate knowledge of the frequencies is available, the spectral estimation described in [197] applies circular von Mises probability distribution on the spectrum.

For *irregularly spaced* or non-uniformly sampled signal, sparse signal recovery methods which leverage on prior information have recently gained attention [17, 108, 114, 181]. Compressed sensing with clustered priors was addressed in [196] where the prior information on the number of clusters and the size of each cluster was assumed to be unknown. In [80], MUSIC was extended to undersampled, irregularly spaced sparse signals in a discrete dictionary, while [122] analyzed the performance of *snapshot*-MUSIC for uniformly sampled signals in a continuous dictionary. Our technique is more general; it applies to irregularly sampled signals in a continuous dictionary, and is, therefore, different from known works on utilizing prior information for spectral estimation of regularly sampled signals.

3.1.2 Problem formulation

In general, the prior information can be available for any of the signal parameters such as amplitude, phase or frequencies. However, in this paper, we restrict the available knowledge to only the frequencies of the signal. We assume that the amplitude and phase information of any of the spectral component is not known, irrespective of the pattern of known frequency information. Our approach is to first analyze the case of a more nebulous prior information, that is the probabilistic priors, followed by an interesting special case of block priors. The case when *some* frequencies are precisely known is considered in the end where, unlike previously considered cases, we recover the signal using the semidefinite program for the primal problem.

We consider a frequency-sparse signal $x[l]$ expressed as a sum of s complex exponentials,

$$x[l] = \sum_{j=1}^{s} c_j e^{i2\pi f_j l} = \sum_{j=1}^{s} |c_j| a(f_j, \phi_j)[l] \ , \ \ l \in \mathcal{N}, \tag{3.1}$$

where $c_j = |c_j| e^{i\phi_j}$ $(i = \sqrt{-1})$ represents the complex coefficient of the frequency $f_j \in [0, 1]$, with amplitude $|c_j| > 0$, phase $\phi_j \in [0, 2\pi)$, and frequency-*atom* $a(f_j, \phi_j)[l] = e^{i(2\pi f_j l + \phi_j)}$. We use the index set $\mathcal{N} = \{l \mid 0 \leq l \leq n-1\}$, where $|\mathcal{N}| = n, n \in \mathbb{N}$, to represent the time samples of the signal. We further suppose that the signal in (3.1) is observed on the index set $\mathcal{M} \subseteq \mathcal{N}$, $|\mathcal{M}| = m \leq n$ where m observations are chosen uniformly at random. Our objective is to recover all the continuous-valued the frequencies with very high accuracy using this undersampled signal.

The signal in (3.1) can be modeled as a positive linear combination of the unit-norm frequency-*atoms* $a(f_j, \phi_j)[l] \in \mathcal{A} \subset \mathbb{C}^n$ where $\mathcal{A}$ is the set of all the frequency-

atoms. These frequency atoms are basic units for synthesizing the frequency-sparse signal. This leads to the following formulation of the *atomic norm* $\|\hat{x}\|_{\mathcal{A}}$ - a sparsity-enforcing analog of ℓ_1 norm for a general atomic set $\mathcal{A}$:

$$\|\hat{x}\|_{\mathcal{A}} = \inf_{c_j, f_j} \left\{ \sum_{j=1}^{s} |c_j| : \hat{x}[l] = \sum_{j=1}^{s} c_j e^{i2\pi f_j l} \ , \ l \in \mathcal{M} \right\}. \tag{3.2}$$

To estimate the remaining $\mathcal{N} \setminus \mathcal{M}$ samples of the signal x, [39] suggests minimizing the atomic norm $\|\hat{x}\|_{\mathcal{A}}$ among all vectors $\hat{x}$ leading to the same observed samples as x. Intuitively, the atomic norm minimization is similar to ℓ_1-minimization being the tightest convex relaxation of the combinatorial ℓ_0-minimization problem. The *primal* convex optimization problem for atomic norm minimization can be formulated as follows,

$$\begin{aligned} &\underset{\hat{x}}{\text{minimize}} \ \ \|\hat{x}\|_{\mathcal{A}} \\ &\text{subject to} \ \ \hat{x}[l] = x[l], \ \ l \in \mathcal{M}. \end{aligned} \tag{3.3}$$

Equivalently, the off-the-grid compressed sensing [168] suggests the following semidefinite characterization for $\|\hat{x}\|_{\mathcal{A}}$:

Definition 3.1. [168] Let T_n denote the $n \times n$ positive semidefinite Toeplitz matrix, $t \in \mathbb{R}^+$, $\mathrm{tr}(\cdot)$ denote the trace operator and $(\cdot)^*$ denote the complex conjugate. Then,

$$\|\hat{x}\|_{\mathcal{A}} = \inf_{T_n, t} \left\{ \frac{1}{2|\mathcal{N}|}\mathrm{tr}(T_n) + \frac{1}{2}t : \begin{bmatrix} [r]T_n & \hat{x} \\ \hat{x}^* & t \end{bmatrix} \succeq 0 \right\}. \tag{3.4}$$

The positive semidefinite Toeplitz matrix T_n is related to the frequency atoms through the following Vandermonde decomposition result by Carathèodory [33]

$$T_n = URU^*, \tag{3.5}$$

where

$$U_{lj} = a(f_j, \phi_j)[l], \tag{3.6}$$

$$R = \mathrm{diag}([b_1, \cdots, b_r]). \tag{3.7}$$

The diagonal elements of R are real and positive, and $r = \mathrm{rank}(T_n)$.

Consistent with this definition, the atomic norm minimization problem for the frequency-sparse signal recovery can now be formulated as a semidefinite program (SDP) with m affine equality constraints:

$$\begin{aligned} &\underset{T_n, \hat{x}, t}{\text{minimize}} \ \ \frac{1}{2|\mathcal{N}|}\mathrm{tr}(T_n) + \frac{1}{2}t \\ &\text{subject to} \ \begin{bmatrix} [r]T_n & \hat{x} \\ \hat{x}^* & t \end{bmatrix} \succeq 0 \\ &\hat{x}[l] = x[l], \ \ l \in \mathcal{M}. \end{aligned} \tag{3.8}$$

When some information about the signal frequencies is known *a priori*, then our goal is to find a signal vector $\hat{x}$ in (3.8) whose frequencies satisfy additional constraints imposed by prior information. In other words, if $\mathcal{C}$ denotes the set of constraints arising due to prior knowledge of frequencies, then our goal is to find the infimum in (3.2) over $f_j \in \mathcal{C}$.

While framing the problem to harness the prior information, a common approach in compressed sensing algorithms is to replace the classical minimization program with its weighted counterpart [114, 181]. However, signals with continuous-valued frequencies do not lead to a direct application of the weighted ℓ_1 approach.

Rather, such an application leads to a fundamental conundrum: the Vandermonde decomposition of positive semidefinite Toeplitz matrices works for general frequencies wherein the frequency atom in (3.6) can freely take any frequency and phase values, and it is not clear how to further tighten the positive semidefinite Toeplitz structure to incorporate the known prior information. Thus, it is non-trivial to formulate a computable convex program that can incorporate general prior information to improve signal recovery.

3.1.3 Probabilistic priors

In the probabilistic prior model, the probability density function of the frequencies is known. Let F be the random variable that describes the signal frequencies. Let the probability density function (pdf) of F be $p_F(f)$. The problem of line spectrum estimation deals with a finite number of signal frequencies in the domain [0, 1]. For example, we can assume $p_F(f)$ to be piecewise constant as follows. Let the domain $[0,1]$ consist of p disjoint subbands such that $[0,1] = \bigcup_{k=1}^{p} \mathcal{B}_k$ where $\mathcal{B}_k$ denotes a subband or a subset of $[0,1]$. Then the restriction $p_F(f)|_{\mathcal{B}_k}$ of $p_F(f)$ to $\mathcal{B}_k$ is a constant. Figure 3.1 illustrates a simple case for $p = 2$, where the line spectrum $X(f)$ of a signal x is non-uniformly sparse over two frequency subbands $\mathcal{B}_1$ and $\mathcal{B}_2 = [0,1]\backslash\mathcal{B}_1$, such that the frequencies f_j, $j = 1, \cdots, s$, occur in the subinterval $\mathcal{B}_2$ more likely than in $\mathcal{B}_1$.

Intuitively, given probabilistic priors, one may think of recovering the signal x

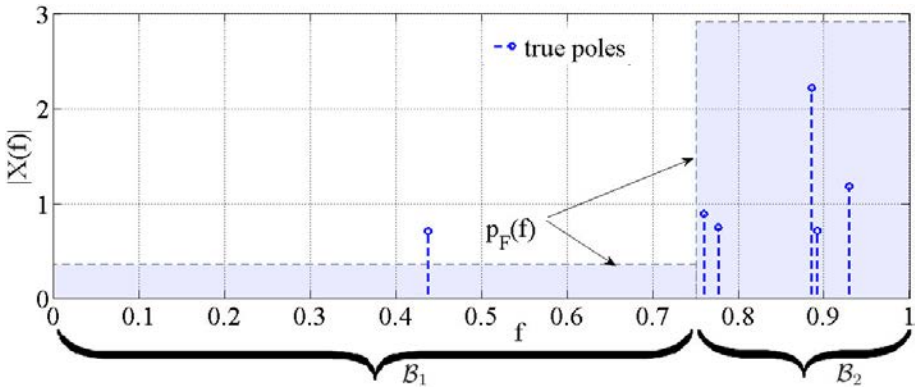


Figure 3.1: The probability density function $p_F(f)$ of the frequencies shown with the location of true frequencies in the spectrum $X(f)$ of the signal $x[l]$.

by minimizing a *weighted atomic norm* given by:

$$\|\hat{x}\|_{\mathbf{w}\mathcal{A}} = \inf_{c_j, f_j} \left\{ \sum_{j=1}^{s} w_j |c_j| : \hat{x}[l] = \sum_{j=1}^{s} c_j e^{i2\pi f_j l} \ , \ l \in \mathcal{M} \right\}, \tag{3.9}$$

where $\mathbf{w} = \{w_1, \cdots, w_s\}$ is the weight vector, each element w_j of which is associated with the probability of occurrence of the corresponding signal frequency f_j. The weight vectors are assigned using a *weight function* $w(f)$. The $w(f)$ is a piecewise constant function in the domain $[0, 1]$ such that the restriction $w(f)|_{\mathcal{B}_k}$ of $w(f)$ to $\mathcal{B}_k$ is a constant. Therefore, $\forall \ \{f_1, \cdots, f_j\} \in \mathcal{B}_k$, we have $w_1 = \cdots = w_j = w(f)|_{f \in \mathcal{B}_k} = w(f_{\mathcal{B}_k})$ (say). The $w(f)$ is a decreasing function of the likelihood of occurrence of frequencies associated with the subband. For example, a subband with a higher pdf of occurrence of frequencies than other subbands is likely to have more frequencies than other subbands. Consequently, such a subband is weighted lightly in comparison to other subbands, in order to promote more frequency occurrence in such a suband.

The problem of line spectral estimation using probabilistic prior can now be

presented as the (primal) optimization problem concerning the weighted atomic norm:

$$\begin{aligned} &\underset{\hat{x}}{\text{minimize}} \ \ \|\hat{x}\|_{\mathbf{w}\mathcal{A}} \\ &\text{subject to} \ \ \hat{x}[j] = x[j], \ \ l \in \mathcal{M}. \end{aligned} \tag{3.10}$$

But we now observe that, unlike weighted ℓ_1 norm [114], a semidefinite characterization of the weighted atomic norm does not evidently result from (3.8). Instead, we propose a new semidefinite program for the weighted atomic norm using theories of positive trigonometric polynomials, by looking at its dual problem. For the standard atomic norm minimization problem (3.3), the dual problem is framed in this manner:

$$\begin{aligned} &\underset{q}{\text{maximize}} \ \ \langle q_{\mathcal{M}}, x_{\mathcal{M}} \rangle_{\mathbb{R}} \\ &\text{subject to} \ \ \|q\|_{\mathcal{A}}^{*} \leq 1 \\ &\qquad\qquad q_{\mathcal{N} \setminus \mathcal{M}} = 0, \end{aligned} \tag{3.11}$$

where $\|\cdot\|^{*}$ represents the dual norm. This dual norm is defined as

$$\|q\|_{\mathcal{A}}^{*} = \sup_{\|\hat{x}\|_{\mathcal{A}} \leq 1} \langle q, \hat{x} \rangle_{\mathbb{R}} = \sup_{f \in [0,1]} |\langle q, a(f,0) \rangle|. \tag{3.12}$$

For the weighted atomic norm minimization, the primal problem (3.10) has only equality constraints. As a result, Slater's condition is satisfied and, therefore, strong duality holds [20]. In other words, solving the dual problem also yields an exact solution to the primal problem. The dual of weighted atomic norm is given by

$$\begin{aligned} \|q\|_{\mathbf{w}\mathcal{A}}^{*} &= \sup_{\|\hat{x}\|_{\mathbf{w}\mathcal{A}} \leq 1} \langle q, \hat{x} \rangle_{\mathbb{R}} = \sup_{\phi \in [0,2\pi], f \in [0,1]} \left\langle q, \frac{1}{w(f)} e^{i\phi} a(f,0) \right\rangle_{\mathbb{R}} \\ &= \sup_{f \in [0,1]} \left| \left\langle q, \frac{1}{w(f)} a(f,0) \right\rangle \right|. \end{aligned} \tag{3.13}$$

The dual problem to (3.10) can be stated hence,

$$\begin{aligned} \underset{q}{\text{maximize}} \;\; & \langle q_{\mathcal{M}}, x_{\mathcal{M}} \rangle_{\mathbb{R}} \\ \text{subject to} \;\; & \|q\|^{*}_{\mathbf{w}\mathcal{A}} \leq 1 \\ & q_{\mathcal{N} \setminus \mathcal{M}} = 0, \end{aligned} \tag{3.14}$$

which by substitution of (3.13) becomes,

$$\begin{aligned} \underset{q}{\text{maximize}} \;\; & \langle q_{\mathcal{M}}, x_{\mathcal{M}} \rangle_{\mathbb{R}} \\ \text{subject to} \;\; & \sup_{f \in [0,1]} \left| \left\langle q, \frac{1}{w(f)} a(f,0) \right\rangle \right| \leq 1 \\ & q_{\mathcal{N} \setminus \mathcal{M}} = 0. \end{aligned} \tag{3.15}$$

Let the probabilistic priors consist of distinct weights for p different frequency sub-bands $\mathcal{B}_k \subset [0,1]$, $k = 1, \cdots, p$ such that $[0,1] = \bigcup_{k=1}^{p} \mathcal{B}_k = \bigcup_{k=1}^{p} [f_{L_k}, f_{H_k}]$, where f_{L_k} and f_{H_k} are, respectively, the lower and upper cut-off frequencies for each of the band $\mathcal{B}_k$ (Figure 3.2). If the probability density function is constant within a frequency band, then the results of the supremums in (3.15) need not depend on the weight functions, and therefore, the inequality constraint in the dual problem in (3.15) can

be expanded as,

$$\begin{aligned}
\underset{q}{\text{maximize}} \quad & \langle q_{\mathcal{M}}, x_{\mathcal{M}} \rangle_{\mathbb{R}} \\
\text{subject to} \quad & \sup_{f \in \mathcal{B}_1} |\langle q, a(f,0) \rangle| \leq w(f_{\mathcal{B}_1}) \\
& \sup_{f \in \mathcal{B}_2} |\langle q, a(f,0) \rangle| \leq w(f_{\mathcal{B}_2}) \\
& \vdots \\
& \sup_{f \in \mathcal{B}_p} |\langle q, a(f,0) \rangle| \leq w(f_{\mathcal{B}_p}) \\
& q_{\mathcal{N} \setminus \mathcal{M}} = 0.
\end{aligned} \tag{3.16}$$

We now map each of the inequality constraints in the foregoing dual problem to a linear matrix inequality, leading to the semidefinite characterization of the weighted atomic norm minimization. We recognize that the constraints in (3.16) imply $Q(f) = \langle q, a(f,0) \rangle$ is a trigonometric polynomial [81] in $f \in \mathcal{B}_k$:

$$Q(f) = \langle q, a(f,0) \rangle = \sum_{l=0}^{n-1} q_l e^{-i2\pi fl}. \tag{3.17}$$

Since $e^{-i2\pi fl} = \cos 2\pi fl + i \sin 2\pi fl$, $Q(f)$ is a trigonometric polynomial. The weights $w(f_{B_k})$ are nonnegative; therefore, the inequalities in (3.16) imply $1 - |Q(f)|^2$ is nonnegative on the unit circle ($|e^{-i2\pi fl}| = 1$). So, we can use the theory of positive trigonometric polynomial from [74] to transform (3.16) into linear matrix inequalities. Such a polynomial can be parameterized by a particular type of positive semidefinite matrix [74, 81]. Thus, we can transform the polynomial inequalities in (3.16), to linear matrix inequalities, using the results of positive trigonometric polynomials from [74], as detailed immediately below.

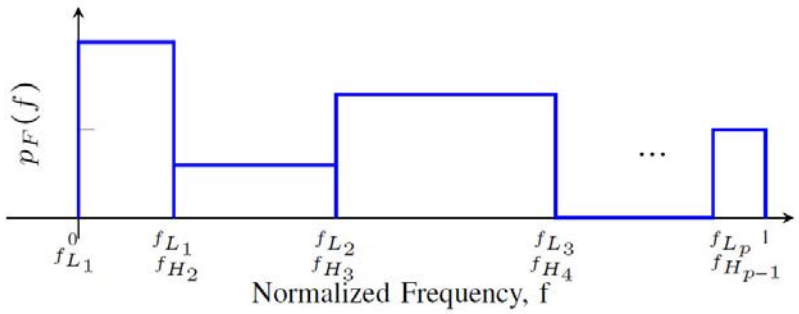


Figure 3.2: The individual frequencies of spectrally parsimonious signal are assumed to lie in known frequency subbands within the normalized frequency domain $[0, 1]$. We assume that all subbands are non-overlapping so that when $f_{H_{k-1}} = f_{L_k}$, then $\mathcal{B}_{k-1} = [f_{L_{k-1}}, f_{H_{k-1}}]$ and $\mathcal{B}_k = (f_{L_k}, f_{H_k}]$.

3.1.3.1 Gram matrix parametrization

A trigonometric polynomial $R(z) = \sum_{k=-(n-1)}^{n-1} r_k z^{-k}$, which is also nonnegative on the entire unit circle, can be parametrized using a positive semidefinite, Hermitian matrix $\boldsymbol{G}$ (called the *Gram* matrix) that identifies the polynomial coefficients r_k as a function of its elements [74, p. 23]:

$$r_k = tr[\boldsymbol{\Theta}_k \boldsymbol{G}], \tag{3.18}$$

where $\boldsymbol{\Theta}_k$ is an elementary Toeplitz matrix with ones on its kth diagonal and zeros elsewhere. Here, $k = 0$ corresponds to the main diagonal, and k takes positive and negative values for upper and lower diagonals respectively.

For the trigonometric polynomial that is nonnegative only over an arc of the unit circle, we have the following theorem:

Theorem 3.1. *[74, p. 12] A trigonometric polynomial*

$$R(z) = \sum_{k=-(n-1)}^{n-1} r_k z^{-k}, \quad r_{-k} = r_k^*, \tag{3.19}$$

where $R \in \mathbb{C}_{n-1}[z]$ *for which* $R(\omega) \geq 0$*, for any* $z = e^{i\omega}$*,* $\omega \in [\omega_L, \omega_H] \subset [-\pi, \pi]$*, can be expressed as*

$$R(z) = F(z)F^*(z^{-1}) + D_{\omega_L\omega_H}(z).G(z)G^*(z^{-1}), \tag{3.20}$$

where $F(z)$*, and* $G(z)$ *are causal polynomials with complex coefficients, of degree at most* $n-1$ *and* $n-2$*, respectively. The polynomial*

$$D_{\omega_L\omega_H}(z) = d_1 z^{-1} + d_0 + d_1^* z \tag{3.21}$$

where

$$d_0 = -\frac{\alpha\beta + 1}{2} \tag{3.22}$$

$$d_1 = \frac{1 - \alpha\beta}{4} + j\frac{\alpha + \beta}{4} \tag{3.23}$$

$$\alpha = \tan\frac{\omega_L}{2} \tag{3.24}$$

$$\beta = \tan\frac{\omega_H}{2}, \tag{3.25}$$

is defined such that $D_{\omega_L\omega_H}(\omega)$ *is nonnegative for* $\omega \in [\omega_L, \omega_H]$ *and negative on its complementary.*[1]

Since $F(z)$ and $G(z)$ are causal polynomials, the products $F(z)F^*(z^{-1})$ and $G(z)G^*(z^{-1})$ are positive trigonometric polynomials that can each be separately parameterized with Gram matrices $\boldsymbol{G}_1$ and $\boldsymbol{G}_2$ respectively.

Proposition 3.2. *A trigonometric polynomial* R *in (3.19) that is nonnegative on the arc* $[\omega_L, \omega_H] \subset [-\pi, \pi]$ *or, alternatively, the subband* $[f_L, f_H] \subset [0, 1]$*, can be*

[1] cf. Errata to [74] available online. The 2007 print edition of [74] has an error in the expression (3.23).

parameterized using the Gram matrices $\boldsymbol{G}_1 \in \mathbb{C}^{n\times n}$ *and* $\boldsymbol{G}_2 \in \mathbb{C}^{(n-1)\times(n-1)}$ *as follows:*

$$\begin{aligned} r_k &= Tr[\boldsymbol{\Theta}_k \boldsymbol{G}_1] + Tr[(d_1\boldsymbol{\Theta}_{k-1} + d_0\boldsymbol{\Theta}_k + d_1^*\boldsymbol{\Theta}_{k+1})\cdot \boldsymbol{G}_2] \\ &\triangleq \mathcal{L}_{k,f_L,f_H}(\boldsymbol{G}_1, \boldsymbol{G}_2), \end{aligned} \tag{3.26}$$

where we additionally require the elementary Toeplitz matrix $\boldsymbol{\Theta}_k$ *in the second argument to be a nilpotent matrix of order* $n-k$ *for* $|k| > 0$*. The translation of frequencies between the two domains is given by* ($f_L < f_H$)*:*

$$\omega_L = \begin{cases} 2\pi f_L & : 0 \le f_L \le 0.5 \\ 2\pi(f_L - 1) & : 0.5 < f_L < 1 \end{cases} \tag{3.27}$$

$$\omega_H = \begin{cases} 2\pi f_H & : 0 < f_H \le 0.5 \\ 2\pi(f_H - 1) & : 0.5 < f_H \le 1 \end{cases}. \tag{3.28}$$

We use the notation $\mathcal{L}_{k,f_L,f_H}(\boldsymbol{G}_1, \boldsymbol{G}_2)$ to concisely denote

$$Tr[\boldsymbol{\Theta}_k \boldsymbol{G}_1] + Tr[(d_1\boldsymbol{\Theta}_{k-1} + d_0\boldsymbol{\Theta}_k + d_1^*\boldsymbol{\Theta}_{k+1})\cdot \boldsymbol{G}_2]$$

, where $\boldsymbol{G}_1 \succeq 0$, $\boldsymbol{G}_2 \succeq 0$ and Θ_k is an elementary Toeplitz matrix with ones on the k-th diagonal and zeros otherwise. Thus $r_k = \mathcal{L}_{k,f_L,f_H}(\boldsymbol{G}_1, \boldsymbol{G}_2)$ is a function of $\boldsymbol{G}_1$, $\boldsymbol{G}_2$ and k. We further remark that the constants d_0 and d_1 depend only on the given subband $[f_L, f_H]$. When it is convenient to refer to the subband in $[-\pi, \pi]$ domain, we change the notation to $\mathcal{L}_{k,\omega_L,\omega_H}(\boldsymbol{G}_1, \boldsymbol{G}_2)$. For a given subband, only the Toeplitz matrices $\boldsymbol{\Theta}_{k-1}$, $\boldsymbol{\Theta}_k$, and $\boldsymbol{\Theta}_{k+1}$ vary for different coefficients of the polynomial. The

corresponding notation for (3.18) would be $r_k = \mathcal{L}_k(\boldsymbol{G})$, where $\boldsymbol{G} \geq 0$. Here, we omit the band from the subscript to imply the band is $[0,1]$ or $[-\pi,\pi]$.

Proof of Proposition 3.2. Let $F(z)$ and $G(z)$ be causal polynomials such that, $F(z) = \mathbf{f}^T\psi(z^{-1})$, and $G(z) = \mathbf{g}^T\phi(z^{-1})$, where $\mathbf{f} = \begin{bmatrix} f_0 & f_1 & \dots & f_{n-1} \end{bmatrix}^T \in \mathbb{C}^n$, and $\mathbf{g} = \begin{bmatrix} g_0 & g_1 & \cdots & g_{n-2} \end{bmatrix}^T \in \mathbb{C}^{n-1}$ are vectors of coefficients of the causal polynomials $F(z)$ and $G(z)$ respectively, and

$$\psi(z^{-1}) = \begin{bmatrix} 1 & z^{-1} & \dots & z^{-(n-1)} \end{bmatrix}^T$$

, and

$$\phi(z^{-1}) = \begin{bmatrix} 1 & z^{-1} & \dots & z^{-(n-2)} \end{bmatrix}^T$$

, are the canonical basis vectors of the corresponding polynomials. Let

$$R(z) = \sum_{k=-(n-1)}^{(n-1)} r_k z^{-k}, \;\; r_{-k} = r_k^*$$

$$A(z) = \sum_{k=-(n-1)}^{n-1} a_k z^{-k} = F(z)F^*(z^{-1}), \;\; a_{-k} = a_k^*$$

$$B(z) = \sum_{k=-(n-2)}^{n-2} b_k z^{-k} = G(z)G^*(z^{-1}), \;\; b_{-k} = b_k^*$$

$$\tilde{B}(z) = \sum_{k=-(n-1)}^{n-1} \tilde{b}_k z^{-k} = D_{\omega_L\omega_H}(z).G(z)G^*(z^{-1}), \;\; \tilde{b}_{-k} = \tilde{b}_k^*.$$

From the above, $r_k = a_k + \tilde{b}_k$. Let $\boldsymbol{G}_1 \in \mathbb{C}^{n\times n}$ and $\boldsymbol{G}_2 \in \mathbb{C}^{(n-1)\times(n-1)}$ be the Gram matrices. Then, as shown in (3.18), the parameterization process yields, $a_k = tr[\boldsymbol{\Theta}_k\boldsymbol{G}_1]$. Also, by definition, if the Gram matrix $\boldsymbol{G}_2$ is associated with a trigonometric poly-

nomial $B(z)$, then we have

$$B(z) = \phi^T(z^{-1}) \cdot \boldsymbol{G}_2 \cdot \phi(z) = Tr[\phi(z) \cdot \phi^T(z^{-1}) \cdot \boldsymbol{G}_2]$$

$$= Tr[\Phi(z) \cdot \boldsymbol{G}_2], \tag{3.29}$$

where

$$\Phi(z) = \begin{bmatrix} 1 \\ z \\ \vdots \\ z^{n-2} \end{bmatrix} \begin{bmatrix} 1 & z^{-1} & \dots & z^{-(n-2)} \end{bmatrix} = \begin{bmatrix} 1 & z^{-1} & \dots & z^{-(n-2)} \\ z & 1 & \ddots & z^{-(n-3)} \\ \vdots & \ddots & \ddots & \vdots \\ z^{n-2} & z^{n-3} & \dots & 1 \end{bmatrix}.$$

This leads to the following expressions:

$$\Phi(z) = \sum_{k=-(n-2)}^{n-2} \boldsymbol{\Theta}_k z^{-k}, \tag{3.30}$$

$$z^{-1}\Phi(z) = z^{-1} \sum_{k=-(n-2)}^{n-2} \boldsymbol{\Theta}_k z^{-k} = \sum_{k=-(n-3)}^{n-1} \boldsymbol{\Theta}_{k-1} z^{-k}, \tag{3.31}$$

$$z\Phi(z) = z \sum_{k=-(n-2)}^{n-2} \boldsymbol{\Theta}_k z^{-k} = \sum_{k=-(n-1)}^{n-3} \boldsymbol{\Theta}_{k+1} z^{-k}. \tag{3.32}$$

Substitution of (3.30)-(3.32) in (3.29) gives the following matrix-parametric expression,

$$\tilde{B}(z) = (d_1 z^{-1} + d_0 + d_1^* z) Tr[\Phi(z) \cdot \boldsymbol{G}_2]$$

$$= Tr[(d_1 z^{-1}\Phi(z) + d_0\Phi(z) + d_1^* z\Phi(z)) \cdot \boldsymbol{G}_2]$$

$$= Tr[(d_1 \sum_{k=-(n-3)}^{n-1} \boldsymbol{\Theta}_{k-1} z^{-k} + d_0 \sum_{k=-(n-2)}^{n-2} \boldsymbol{\Theta}_k z^{-k}$$

$$+ d_1^* \sum_{k=-(n-1)}^{n-3} \boldsymbol{\Theta}_{k+1} z^{-k}) \cdot \boldsymbol{G}_2]$$

$$= \sum_{k=-(n-1)}^{n-1} Tr[(d_1\boldsymbol{\Theta}_{k-1} + d_0\boldsymbol{\Theta}_k + d_1^*\boldsymbol{\Theta}_{k+1}) \cdot \boldsymbol{G}_2] z^{-k}.$$

Then,

$$\tilde{b}_k = Tr[(d_1\boldsymbol{\Theta}_{k-1} + d_0\boldsymbol{\Theta}_k + d_1^*\boldsymbol{\Theta}_{k+1}) \cdot \boldsymbol{G}_2]. \tag{3.33}$$

Substitution of matrix parameterizations of a_k and $\tilde{b}_k$ in the expression of r_k completes the proof. □

The dual polynomial $Q(f)$ in (3.17) is nonnegative on multiple non-overlapping intervals, and can therefore be parameterized by as many different pairs of Gram matrices $\{\boldsymbol{G}_1, \boldsymbol{G}_2\}$ as the number of subbands p. In the following subsection, we relate this parametrization to the corresponding probabilistic weights of the subbands.

3.1.3.2 SDP formulation

Based on the Bounded Real Lemma [74, p. 127] (which, in turn, is based on Theorem 3.1), a positive trigonometric polynomial constraint of the type $|R(\omega)| \leq 1$ can be expressed as a linear matrix inequality [74, p. 143]. Stating this result for the dual polynomial constraint over a single frequency band, such as those in (3.16), we have

$$\sup_{f \in [f_L, f_H]} |\langle q, a(f,0)\rangle| \leq \gamma, \tag{3.34}$$

if and only if there exist positive semidefinite *Gram* matrices $\boldsymbol{G}_1 \in \mathbb{C}^{n\times n}$ and $\boldsymbol{G}_2 \in \mathbb{C}^{(n-1)\times(n-1)}$ such that,

$$\gamma^2 \delta_k = \mathcal{L}_{k,\omega_L,\omega_H}(\boldsymbol{G}_1, \boldsymbol{G}_2), \;\; k \in \mathcal{H}$$
$$\begin{bmatrix} \boldsymbol{G}_1 & q \\ q^* & 1 \end{bmatrix} \geq 0, \tag{3.35}$$

where $\mathcal{H}$ is a halfspace, $\delta_0 = 1$, and $\delta_k = 0$ if $k \neq 0$. Here, $\mathcal{L}_{k,\omega_L,\omega_H}(\boldsymbol{G}_1, \boldsymbol{G}_2)$ represents the trace and Gram pair parameterization. There are a total of n coefficients; so,

the halfspace reference in (3.35) implies that k takes values from 0 to $n-1$, leading to n equalities of the type $\gamma^2\delta_k = \mathcal{L}_{k,\omega_L,\omega_H}(\boldsymbol{G}_1, \boldsymbol{G}_2)$. This linear matrix inequality representation using positive semidefinite matrix $\boldsymbol{G}_1$ paves way for casting the new dual problem in (3.16) as a semidefinite program. This above formulation shows that we have changed the inequality form in the convex optimization problem to an equality form allowing semidefinite programming for the weighted atomic norm minimization.

If the cutoff-frequencies ω_L or ω_H (in $[-\pi, \pi]$ domain) are equal to $\pm\pi$, then we can write $[\omega_L, \omega_H] = [\omega_L' + \tau, \omega_H' + \tau]$ such that $[\omega_L', \omega_H'] \subset [-\pi, \pi]$. For the translated subband $[\omega_L', \omega_H']$, let the corresponding subband in the domain $[0, 1]$ be $[f_L', f_H']$. Then, the LMI formulation given by (3.26) becomes valid for this subband. However, the polynomial q is now evaluated in the domain $e^{-i\omega}e^{-i\tau}$ instead of $e^{-i\omega}$. The SDP for this frequency translation employs a scaled version of LMI in (3.35),

$$\begin{aligned} \delta_k &= \mathcal{L}_{k,f_L',f_H'}(\boldsymbol{G}_1, \boldsymbol{G}_2), \;\; k \in \mathcal{H} \\ &\begin{bmatrix} \boldsymbol{G}_1 & \frac{1}{\gamma}\tilde{q}_\tau \\ \frac{1}{\gamma}\tilde{q}_\tau^* & 1 \end{bmatrix} \succeq 0, \end{aligned} \tag{3.36}$$

where

$$\tilde{q}_\tau[j] = q[j]e^{-i\tau j}. \tag{3.37}$$

We now state the semidefinite program for weighted atomic norm minimization with the probabilistic priors. We use the LMI representation for each of the inequality constraints in (3.16) as follows:

$$
\begin{aligned}
&\underset{\substack{q,\\ \boldsymbol{G}_{11},\boldsymbol{G}_{12},\cdots,\boldsymbol{G}_{1p},\\ \boldsymbol{G}_{21},\boldsymbol{G}_{22},\cdots,\boldsymbol{G}_{2p}}}{\text{maximize}} \quad \langle q_{\mathcal{M}}, x_{\mathcal{M}} \rangle_{\mathbb{R}} \\
&\text{subject to} \quad q_{\mathcal{N}\setminus\mathcal{M}} = 0 \qquad (3.38)\\
&\qquad \delta_{k_1} = \mathcal{L}_{k_1, f_{L_1}', f_{H_1}'}(\boldsymbol{G}_{11}, \boldsymbol{G}_{21}),\\
&\qquad\qquad k_1 = 0, \cdots, (n-1)\\
&\qquad \begin{bmatrix} \boldsymbol{G}_{11} & \frac{1}{w_1}\tilde{q}_{\tau_1} \\ \frac{1}{w_1}\tilde{q}^*_{\tau_1} & 1 \end{bmatrix} \succeq 0,\\
&\qquad \delta_{k_2} = \mathcal{L}_{k_2, f_{L_2}', f_{H_2}'}(\boldsymbol{G}_{12}, \boldsymbol{G}_{22}),\\
&\qquad\qquad k_2 = 0, \cdots, (n-1)\\
&\qquad \begin{bmatrix} \boldsymbol{G}_{12} & \frac{1}{w_2}\tilde{q}_{\tau_2} \\ \frac{1}{w_2}\tilde{q}^*_{\tau_2} & 1 \end{bmatrix} \succeq 0,\\
&\qquad\qquad \vdots\\
&\qquad \delta_{k_p} = \mathcal{L}_{k_p, f_{L_p}', f_{H_p}'}(\boldsymbol{G}_{1p}, \boldsymbol{G}_{2p}),\\
&\qquad\qquad k_p = 0, \cdots, (n-1)\\
&\qquad \begin{bmatrix} \boldsymbol{G}_{1p} & \frac{1}{w_p}\tilde{q}_{\tau_p} \\ \frac{1}{w_p}\tilde{q}^*_{\tau_p} & 1 \end{bmatrix} \succeq 0,\\
&\text{where } \tilde{q}_{\tau_k}[j] = q[j]e^{-i\tau_k j}, \quad k = 1, \cdots, p,\\
&\qquad \boldsymbol{G}_{11}, \boldsymbol{G}_{12}, \cdots, \boldsymbol{G}_{1p} \in \mathbb{C}^{n\times n},\\
&\text{and } \boldsymbol{G}_{21}, \boldsymbol{G}_{22}, \cdots, \boldsymbol{G}_{2p} \in \mathbb{C}^{(n-1)\times(n-1)}.
\end{aligned}
$$

Algorithm 3.1 Frequency localization for probabilistic priors

1 Solve the dual problem (3.38) to obtain the optimum solution $q^\star$.
2 Let $\mathcal{F} = \{f_1, \cdots, f_j, \cdots, f_s\}$ be the unknown frequencies of signal x. The unknown frequencies f_j, identify as $|\langle q^\star, a(f_j, 0)\rangle| = w_k$, where $f_j \in \mathcal{B}_k \subseteq [0,1]$. For $f \in (\mathcal{B}_k \setminus \mathcal{F}) \subset [0,1]$, $|\langle q^\star, a(f,0)\rangle| < w_k$.
3 The corresponding complex coefficients can be recovered by solving a system of simultaneous linear equations $\hat{x}[l] - \sum_{j=1}^{s} c_j a(f_j, 0)[l] = 0$.

The unknown frequencies in $\hat{x}$ can be identified by the frequency localization approach [168] based on computing the dual polynomial, that we state for the weighted atomic norm problem in Algorithm 3.1. We state that this characterization of the spectral estimation is a general way to integrate given knowledge about the spectrum. If the engineer is able to locate the signal frequency in a particular subband with a very high degree of certainty, better results can be obtained using the optimization (3.38). Also, information about signal frequency bands is frequently available through previous research and measurements, especially in problems pertaining to communication, power systems and remote sensing. We consider this more practical case in the following section.

3.1.4 Block priors

Of particular interest to spectral estimation are spectrally block sparse signals where certain frequency bands are known to contain all the spectral contents of the signal. Let us assume that all the s frequencies f_j of the spectrally sparse signal x are known *a priori* to lie only in a finite number of non-overlapping frequency bands or intervals within the normalized frequency domain $[0,1]$. Here, the known set $\mathcal{C}$

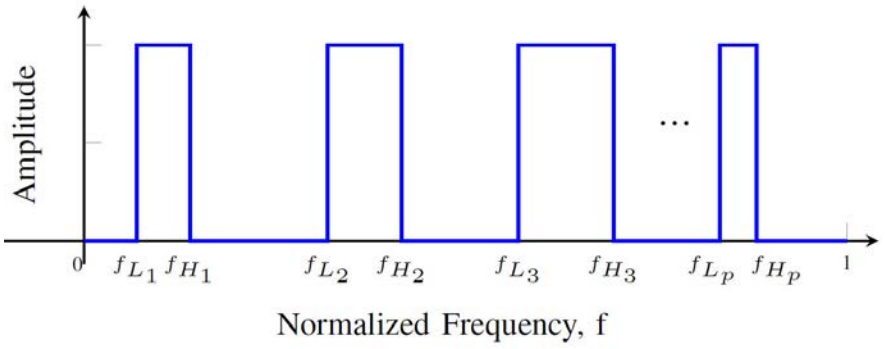


Figure 3.3: The individual frequencies of spectrally sparse signal are assumed to lie in known non-overlapping frequency subbands within the normalized frequency domain $[0, 1]$.

is defined as the set $\mathcal{B}$ of all frequency bands in which signal frequencies are known to reside. The prior information consists of the precise locations of all the frequency bands - the lower and upper cut-off frequencies f_{L_k} and f_{H_k} respectively for each of the band $\mathcal{B}_k$ - as shown in the Figure 3.3. We, therefore, have $f_j \in \mathcal{B}$, $\mathcal{B} = \bigcup_{k=1}^{p} \mathcal{B}_k = \bigcup_{k=1}^{p} [f_{L_k}, f_{H_k}]$, where p is the total number of disjoint bands known *a priori*. This *block prior* problem could easily be considered as a special case of probabilistic priors where the probability of a frequency occurring in known subbands is unity while it is zero for all other subbands. When the frequencies are known to reside in the set of subbands $\mathcal{B}$ *a priori*, we propose to minimize a *constrained* atomic norm $\|\hat{x}\|_{\mathcal{A},\mathcal{B}}$ for perfect recovery of the signal:

$$\|\hat{x}\|_{\mathcal{A},\mathcal{B}} = \inf_{c_j, f_j \in \mathcal{B}} \left\{ \sum_{j=1}^{s} |c_j| : \hat{x}[l] = \sum_{j=1}^{s} c_j e^{i2\pi f_j l} \quad , \quad l \in \mathcal{M} \right\}. \tag{3.39}$$

As noted earlier, to recover all of the off-the-grid frequencies of the signal x given the block priors, the direct extension of a semidefinite program from (3.8) to minimize the constrained atomic norm is non-trivial. We address this problem by working with the

dual problem of the constrained atomic norm minimization, and then transforming the dual problem to an equivalent semidefinite program by using theories of positive trigonometric polynomials. We note that in the case of block priors, (3.12) can be written as $\|q\|^*_{\mathcal{A},\mathcal{B}} = \sup_{f\in\mathcal{B}} |\langle q, a(f,0)\rangle| = \sup_{f\in\mathcal{B}} |Q(f)|$, where $Q(f)$ is the dual polynomial. The primal problem of constrained atomic norm minimization is given by

$$\begin{aligned} &\underset{\hat{x}}{\text{minimize}} \ \ \|\hat{x}\|_{\mathcal{A},\mathcal{B}} \\ &\text{subject to} \ \ \hat{x}[l] = x[l], \ \ l \in \mathcal{M}, \end{aligned} \tag{3.40}$$

and, similar to (3.11), we can formulate the corresponding dual problem as

$$\begin{aligned} &\underset{q}{\text{maximize}} \ \ \langle q_{\mathcal{M}}, x_{\mathcal{M}}\rangle_{\mathbb{R}} \\ &\text{subject to} \ \ q_{\mathcal{N}\setminus\mathcal{M}} = 0 \\ &\qquad\qquad \|q\|^*_{\mathcal{A},\mathcal{B}} \le 1, \end{aligned} \tag{3.41}$$

where $\|q\|^*_{\mathcal{A},\mathcal{B}} = \sup_{f\in\mathcal{B}} |\langle q, a(f,0)\rangle|$. Since $\mathcal{B}$ is defined as a union of multiple frequency bands, the inequality constraint in (3.41) can be expanded to p separate inequality constraints. It can be easily observed that (3.41) is a special case of (3.15) with all the weights being unity and $\mathcal{B} \subseteq [0,1]$ (i. e. the set of bands $\mathcal{B}$ need not necessarily cover the entire frequency range). While framing the semidefinite program for this problem, we use a linear matrix inequality similar to that in (3.35) with $\gamma = 1$ for each of the inequality constraint in (3.41), to cast the dual problem constraint into a semidefinite program. So, when all the frequencies are known to lie in p disjoint frequency bands, then the semidefinite program for

the dual problem in (3.41) can be constructed by using p equality-form constraints:

$$
\begin{aligned}
\underset{\substack{q,\\ \boldsymbol{G}_{11},\boldsymbol{G}_{12},\cdots,\boldsymbol{G}_{1p},\\ \boldsymbol{G}_{21},\boldsymbol{G}_{22},\cdots,\boldsymbol{G}_{2p}}}{\text{maximize}} \quad & \langle q_{\mathcal{M}}, x_{\mathcal{M}} \rangle_{\mathbb{R}} \\
\text{subject to} \quad & q_{\mathcal{N}\setminus\mathcal{M}} = 0 \qquad (3.42) \\
& \delta_{k_1} = \mathcal{L}_{k_1, f_{L_1}, f_{H_1}}(\boldsymbol{G}_{11}, \boldsymbol{G}_{21}), \\
& \qquad k_1 = 0, \cdots, (n-1) \\
& \begin{bmatrix} \boldsymbol{G}_{11} & q \\ q^* & 1 \end{bmatrix} \succeq 0, \\
& \delta_{k_2} = \mathcal{L}_{k_2, f_{L_2}, f_{H_2}}(\boldsymbol{G}_{12}, \boldsymbol{G}_{22}), \\
& \qquad k_2 = 0, \cdots, (n-1) \\
& \begin{bmatrix} \boldsymbol{G}_{12} & q \\ q^* & 1 \end{bmatrix} \succeq 0, \\
& \qquad \vdots \\
& \delta_{k_p} = \mathcal{L}_{k_p, f_{L_p}, f_{H_p}}(\boldsymbol{G}_{1p}, \boldsymbol{G}_{2p}), \\
& \qquad k_p = 0, \cdots, (n-1) \\
& \begin{bmatrix} \boldsymbol{G}_{1p} & q \\ q^* & 1 \end{bmatrix} \succeq 0, \\
& \text{where } \boldsymbol{G}_{11}, \boldsymbol{G}_{12}, \cdots, \boldsymbol{G}_{1p} \in \mathbb{C}^{n \times n}, \\
& \text{and } \boldsymbol{G}_{21}, \boldsymbol{G}_{22}, \cdots, \boldsymbol{G}_{2p} \in \mathbb{C}^{(n-1)\times(n-1)}.
\end{aligned}
$$

In the extreme case when any of the known frequency bands $\mathcal{B}_k$ have ω_{L_k} or ω_{H_k} lying exactly on either $-\pi$ or π, then the dual-polynomial in 3.42 should be appropriately translated as noted in (3.37).

In many applications, the location of *some* of the signal frequencies might be precisely known. One could think of this *known poles* problem as a probabilistic prior problem where the cardinality of *some* sets $\mathcal{B}_k$ is exactly unity (and the associated probability be unity as well), while the remaining frequency subbands have a non-unity probability. However, there are a few differences. For probabilistic priors, the probability distribution function is known for the entire interval $[0, 1]$ while, in case of known poles, the probability distribution of the bands of unknown frequencies is unavailable. Also, unlike block prior formulation, known poles problem does not have zero probability associated with the remaining subbands.

3.1.5 Known poles

We now consider the case when some frequency components are known *a priori* but their corresponding amplitudes and phases are not. Let the index set of all the frequencies be $\mathcal{S}$, $|\mathcal{S}| = s$. Let $\mathcal{P}$ be the index set of all the known frequencies, and $|\mathcal{P}| = p$. Namely, we assume that the signal x contains some known frequencies f_j, $j \in \mathcal{P} \subseteq \mathcal{S}$, $|\mathcal{P}| = p$. For known frequencies, let us denote their *complex* coefficients as d_j and their *phaseless frequency atoms* as $a_j[l] = a(f_j, 0)[l] = e^{i2\pi f_j l}$. We define the *conditional atomic norm* $\|\hat{x}\|_{\mathcal{A},\mathcal{P}}$ for the *known poles* as follows:

$$\|\hat{x}\|_{\mathcal{A},\mathcal{P}} = \inf_{c_j,d_j,f_j} \left\{ \sum_{j=1}^{s-p} |c_j| : \hat{x}[l] = \sum_{j=1}^{s-p} c_j e^{i2\pi f_j l} + \sum_{j=s-p+1}^{s} d_j e^{i2\pi f_j l} \ , \ l \in \mathcal{M} \right\}. \tag{3.43}$$

Unlike previously mentioned *a priori* counterparts of the atomic norm, the semidefinite formulation for $\|\hat{x}\|_{\mathcal{A},\mathcal{P}}$ easily follows from (3.4).

Proposition 3.3. *The conditional atomic norm for a vector $\hat{x}$ is given by*

$$\|\hat{x}\|_{\mathcal{A},\mathcal{P}} = \inf_{T_n,\tilde{x},t,d_j} \left\{ \frac{1}{2|\mathcal{N}|} tr(T_n) + \frac{1}{2}t : \begin{bmatrix} T_n & \tilde{x} \\ \tilde{x}^* & t \end{bmatrix} \succeq 0 \right\}, \tag{3.44}$$

where $\tilde{x}[l] = \hat{x}[l] - \sum_{j\in\mathcal{P}} a_j[l]d_j$ represents the positive combination of complex sinusoids with unknown poles.

Proof of Proposition 3.3. From (3.43), we simply have

$$\tilde{x}[l] = \hat{x}[l] - \sum_{j\in\mathcal{P}} a_j[l]d_j = \sum_{j=1}^{s-p} c_j e^{i2\pi f_j l}$$

, meaning the value of the semidefinite program in (3.44) is same as $\|\tilde{x}\|_{\mathcal{A}} = \|\hat{x}\|_{\mathcal{A},\mathcal{P}}$. □

The conditional atomic norm minimization problem can be posed as the following semidefinite formulation in a similar way as in (3.8):

$$\begin{aligned} &\underset{T_n,\hat{x},\tilde{x},t,d_j}{\text{minimize}} \ \frac{1}{2|\mathcal{N}|}\text{tr}(T_n) + \frac{1}{2}t \\ &\text{subject to} \begin{bmatrix} T_n & \tilde{x} \\ \tilde{x}^* & t \end{bmatrix} \succeq 0 \\ &\hat{x}[l] = x[l], \ \ l \in \mathcal{M} \\ &\hat{x}[l] = \tilde{x}[l] + \sum_{j\in\mathcal{P}} a_j[l]d_j, \ \ l \in \mathcal{M}. \end{aligned} \tag{3.45}$$

Algorithm 3.2 *Known poles* algorithm

1 Solve the semidefinite program (3.45) to obtain $\tilde{x}$.
2 Solve the following dual problem to obtain the optimum solution $q^\star$

$$\begin{aligned} &\underset{q}{\text{maximize}} \ \langle q, \tilde{x} \rangle_{\mathbb{R}} \\ &\text{subject to} \ \|q\|_{\mathcal{A}}^* \leq 1 \\ &q[l] = 0, \ \ l \in \mathcal{N} \setminus \mathcal{M}. \end{aligned} \tag{3.46}$$

3 The unknown frequencies f_j, $j \in \mathcal{P}$, identify as $|\langle q^\star, a_j \rangle| = 1$. For $j \notin \mathcal{S} \setminus \mathcal{P}$, $|\langle q^\star, a_j \rangle| < 1$.
4 Solve the following system of simultaneous linear equations to recover the complex coefficients of unknown frequencies: $\tilde{x}[l] - \sum_{j \in \mathcal{S} \setminus \mathcal{P}} c_j a_j[l] = 0$.

The $\tilde{x}$ can be viewed as the signal filtered of the *known poles*. The remaining unknown frequencies in $\tilde{x}$ can be identified by the frequency localization approach that we restate for $\tilde{x}$ in Algorithm 3.2.

3.1.6 Performance analysis

To identify the true frequencies of the signal from the solution of the dual problem, we now establish the conditions for finding the dual-certificate of support when prior information is available. We additionally show that the dual polynomial requirements can be slackened if the prior information gives the approximate location of each of the signal frequencies. We further put our result in the context of minimum number of signal samples required for the reconstruction of the signal x.

Since the primal problem (3.10) has only equality constraints, Slater's condition is satisfied. As a consequence, strong duality holds [20]. This allows us to present the dual-certificate of support for the optimizer of (3.10). In the following theorems, $\text{sign}(c_j) = {}^{c_j}/_{|c_j|}$, and $\text{Re}(\cdot)$ denotes the real part (of a complex number).

Theorem 3.4. *Let the set of atoms* $\{a_{\mathcal{M}}(f_1,0),\cdots,a_{\mathcal{M}}(f_s,0)\}$ *supported on subset* $\mathcal{M}$ *of* $\mathcal{N}$ *be linearly independent. Then,* $\hat{x} = x$ *is the unique solution to the primal problem (3.10), if there exists a polynomial*

$$Q(f) = \langle q, a(f,0)\rangle = \sum_{l=0}^{n-1} q_l e^{-i2\pi f l}, \tag{3.47}$$

such that

$$Q(f_j) = w_k sign(c_j), \forall f_j \in \mathcal{B}_k \subseteq [0,1] \tag{3.48}$$

$$|Q(f)| < w_k, \forall f \in (\mathcal{B}_k \setminus \mathcal{F}) \subset [0,1] \tag{3.49}$$

$$q_{\mathcal{N}\setminus\mathcal{M}} = 0. \tag{3.50}$$

Proof of Theorem 3.4. The proof follows from the dual polynomial for the standard atomic norm minimization problem. We refer the reader to [168] for details. Briefly, it can be concluded that strong duality holds and we have $\langle q_{\mathcal{M}}, x_{\mathcal{M}}\rangle_{\mathbb{R}} = \sum_{j=1}^{s} w_j|c_j| = \|x\|_{\mathbf{w}\mathcal{A}}$, where the vector q satisfies the conditions (3.48), (3.49), and (3.50), and is dual feasible. As for the uniqueness, let $x^\dagger[l] = \sum_j c_j^\dagger e^{i2\pi f_j^\dagger l}$, $l \in \mathcal{M}$, be an alternative minimizer of (3.10) such that $x^\dagger$ contains frequencies outside the set $\mathcal{F}$ of oracle

frequencies. Then,

$$\begin{aligned}
\|x\|_{\mathbf{w}\mathcal{A}} &= \langle q_{\mathcal{M}}, x_{\mathcal{M}} \rangle_{\mathbb{R}} = \langle q_{\mathcal{M}}, x^{\dagger}_{\mathcal{M}} \rangle_{\mathbb{R}} = \langle q_{\mathcal{M}}, \sum_j c_j^{\dagger} e^{i2\pi f_j^{\dagger} l} \rangle_{\mathbb{R}} \\
&= \left\langle q_{\mathcal{M}}, \sum_k \sum_{f_j \in \mathcal{F} \subset \mathcal{B}_k} c_j^{\dagger} e^{i2\pi f_j l} + \sum_k \sum_{f_h^{\dagger} \in (\mathcal{B}_k \setminus \mathcal{F})} c_h^{\dagger} e^{i2\pi f_h^{\dagger} l} \right\rangle_{\mathbb{R}} \\
&< \sum_{f_j \in \mathcal{F} \subset \mathcal{B}_k} w_j |c_j^{\dagger}| + \sum_{f_h^{\dagger} \in (\mathcal{B}_k \setminus \mathcal{F})} w_h |c_h^{\dagger}| \\
&= \|x^{\dagger}\|_{\mathbf{w}\mathcal{A}},
\end{aligned}$$

resulting in a contradiction that $x^{\dagger}$ is not a minimizer of (3.10). If $x^{\dagger}$ contains only the oracle frequencies and the same sign pattern $c_j/|c_j|$ as that of x, then $x^{\dagger}$ also has the same complex coefficients as x since the set $\{a_{\mathcal{M}}(f_1, 0), \cdots, a_{\mathcal{M}}(f_s, 0)\}$ is linearly independent. Therefore, the optimal solution is unique. □

Note that at least $|\mathcal{M}| = s$ observations are required to guarantee the linear independence of the set $\{a_{\mathcal{M}}(f_1, 0), \cdots, a_{\mathcal{M}}(f_s, 0)\}$. Moreover, following similar arguments in the proof of Proposition 3.7, if frequencies f_1, f_2,..., and f_M are drawn uniformly at random from $[0, 1]$, for any $\mathcal{M}$ such that $|\mathcal{M}|$=s, the set of atoms $\{a_{\mathcal{M}}(f_1, 0), \cdots, a_{\mathcal{M}}(f_s, 0)\}$ are linearly independent with probability 1. As a corollary to Theorem 3.4, we can arrive at the dual polynomial for the block prior problem as follows.

Corollary 3.5. *The $\hat{x} = x$ is the unique solution to the primal problem (3.40), if*

there exists a polynomial $Q(f)$ *such that*

$$Q(f_j) = sign(c_j), \forall f_j \in \mathcal{F} \subset \mathcal{B} \tag{3.51}$$

$$|Q(f)| < 1, \forall f \in (\mathcal{B} \setminus \mathcal{F}) \tag{3.52}$$

$$q_{\mathcal{N}\setminus\mathcal{M}} = 0. \tag{3.53}$$

When the prior information is available to such a generous extent that each of the individual frequencies are known within close boundaries, as we present next, an interesting consequence of this relaxation is that the number of samples required to reconstruct the signal could be bounded.

Theorem 3.6. *Let the signal* x *as in (3.1) be sampled on a subset* $\mathcal{M}$ *of* $\mathcal{N}$*. If there exists a polynomial* $Q(f)$ *such that* $\forall f_j \in \mathcal{F} \subset \mathcal{B}$*,*

$$Q(f_j) = sign(c_j) \tag{3.54}$$

$$Q'(f_j) = \sum_{l=0}^{n-1} l q_l e^{-i2\pi f_j l} = 0 \tag{3.55}$$

$$Q''(f_j) = \sum_{l=0}^{n-1} -(2\pi l)^2 q_l e^{-i2\pi f_j l} = -sign(Re(c_j)), \tag{3.56}$$

and, if each of the frequencies is known within a sufficiently small frequency subband, then $\hat{x} = x$ *is the unique optimizer of the primal problem (3.40). Further, assuming* $f_j s$ *are distributed uniformly at random in* $[0, 1]$*, such a dual polynomial exists with probability* 1 *when* $m \geq 3s$*.*

Proof of Theorem 3.6. The polynomial that we seek can be written as $Q(f) = Q_R(f) + iQ_I(f)$, where $Q_R(f)$ and $Q_I(f)$ are the real and imaginary parts respectively. As per

Theorem 3.4, $Q(f)$ should also satisfy the conditions (3.48) and (3.49). Therefore, (3.54) is a restatement of (3.48) as follows:

$$Q(f_j) = \sum_{l=0}^{n-1} q_l e^{-i2\pi f_j l} = \text{sign}(c_j) = \frac{c_j}{|c_j|} \quad \forall f_j \in \mathcal{F} \subset \mathcal{B}. \tag{3.57}$$

For the dual polynomial to achieve an extremum at $f_j \in \mathcal{F} \subset \mathcal{B}$ as specified by (3.49), the following is a sufficient condition for its first derivative leading to (3.55):

$$Q'(f_j) = \sum_{l=0}^{n-1} -i2\pi l q_l e^{-i2\pi f_j l} = 0 \quad \forall f_j \in \mathcal{F} \subset \mathcal{B}. \tag{3.58}$$

The condition for a maximum at $f_j \in \mathcal{F} \subset \mathcal{B}$ requires the second derivative $|Q(f_j)|''$ to be strictly negative. We have,

$$\begin{aligned} |Q(f_j)|'' = &-\frac{[Q_R(f_j)Q_R'(f_j) + Q_I(f_j)Q_I'(f_j)]^2}{|Q(f_j)|^3} \\ &+ \frac{|Q'(f_j)|^2 + Q_R(f_j)Q_R''(f_j) + Q_I(f_j)Q_I''(f_j)}{|Q(f_j)|} \\ &\forall f_j \in \mathcal{F} \subset \mathcal{B}. \end{aligned} \tag{3.59}$$

Therefore, for $|Q(f_j)|''$ to be strictly negative, it is sufficient to require,

$$|Q'(f_j)|^2 + Q_R(f_j)Q_R''(f_j) + Q_I(f_j)Q_I''(f_j) < 0 \quad \forall f_j \in \mathcal{F} \subset \mathcal{B}. \tag{3.60}$$

Under the condition (3.60), when the frequencies f_j are known to lie in a very small frequency band $\mathcal{B}_k$ such that $(f_{H_k} - f_{L_k}) \ll 1$, then the polynomial constraints are valid within such a sufficiently small interval.

To satisfy the constraint (3.60), we impose an additional constraint that requires $Q_I''(f_j)$ to vanish, reducing (3.60) to

$$Q_R(f_j)Q_R''(f_j) < 0 \quad \forall f_j \in \mathcal{F} \subset \mathcal{B}. \tag{3.61}$$

Using the definition of dual polynomial from (3.47), we can now cast (3.61) as the condition (3.56).

Let $x_{f_j} = e^{i2\pi f_j}$. We show that the linear system (3.54), (3.55), and (3.56) results in a unique solution, given at least $3s$ equations as follows:

$$\underbrace{\begin{bmatrix} x_{f_1}^{l_0} & x_{f_1}^{l_1} & \cdots & x_{f_1}^{l_{3s-1}} \\ l_0 x_{f_1}^{l_0} & l_1 x_{f_1}^{l_1} & \cdots & l_{3s-1} x_{f_1}^{l_{3s-1}} \\ -(2\pi l_0)^2 x_{f_1}^{l_0} & -(2\pi l_1)^2 x_{f_1}^{l_1} & \cdots & -(2\pi l_{3s-1})^2 x_{f_1}^{l_{3s-1}} \\ x_{f_2}^{l_0} & x_{f_2}^{l_1} & \cdots & x_{f_2}^{l_{3s-1}} \\ l_0 x_{f_2}^{l_0} & l_1 x_{f_2}^{l_1} & \cdots & l_{3s-1} x_{f_2}^{l_{3s-1}} \\ -(2\pi l_0)^2 x_{f_2}^{l_0} & -(2\pi l_1)^2 x_{f_2}^{l_1} & \cdots & -(2\pi l_{3s-1})^2 x_{f_2}^{l_{3s-1}} \\ \vdots & \vdots & \ddots & \vdots \\ x_{f_s}^{l_0} & x_{f_s}^{l_1} & \cdots & x_{f_s}^{l_{3s-1}} \\ l_0 x_{f_s}^{l_0} & l_1 x_{f_s}^{l_1} & \cdots & l_{3s-1} x_{f_s}^{l_{3s-1}} \\ -(2\pi l_0)^2 x_{f_s}^{l_0} & -(2\pi l_1)^2 x_{f_s}^{l_1} & \cdots & -(2\pi l_{3s-1})^2 x_{f_s}^{l_{3s-1}} \end{bmatrix}}_{= \boldsymbol{A}} \begin{bmatrix} q_{l_0} \\ q_{l_0} \\ q_{l_0} \\ q_{l_1} \\ q_{l_1} \\ q_{l_1} \\ \vdots \\ q_{l_{3s-1}} \\ q_{l_{3s-1}} \\ q_{l_{3s-1}} \end{bmatrix}$$

$$= \left[\tfrac{c_1}{|c_1|} \; 0 \; -\text{sign}(\text{Re}(c_1)) \; \tfrac{c_2}{|c_2|} \; 0 \; -\text{sign}(\text{Re}(c_2)) \; \cdots \; \tfrac{c_s}{|c_s|} \; 0 \; -\text{sign}(\text{Re}(c_s)) \right]^T, \tag{3.62}$$

where $l_0, l_1, \cdots, l_{3s-1}$ are the indices of the samples of the signal x. Proposition 3.7 completes the proof by showing that the system matrix $\boldsymbol{A}$ in (3.62) is invertible with probability 1, provided the frequencies in the set $\mathcal{F} = \{f_1, \cdots, f_j, \cdots, f_s\}$ are distributed uniformly at random. □

Proposition 3.7. *Let $\mathcal{M} = \{l_0, l_1, \cdots, l_{3s-1}\}$ be the set of indices for $3s$ samples of the*

signal x. *Let* $h_{f_j} = e^{i2\pi f_j}$, *then the* $3s \times 3s$ *matrix*

$$\boldsymbol{A}_s = \begin{bmatrix} h_{f_1}^{l_0} & h_{f_1}^{l_1} & \cdots & h_{f_1}^{l_{3s-1}} \\ l_0 \cdot h_{f_1}^{l_0} & l_1 \cdot h_{f_1}^{l_1} & \cdots & l_{3s-1} \cdot h_{f_1}^{l_{3s-1}} \\ l_0^2 \cdot h_{f_1}^{l_0} & l_1^2 \cdot h_{f_1}^{l_1} & \cdots & l_{3s-1}^2 \cdot h_{f_1}^{l_{3s-1}} \\ h_{f_2}^{l_0} & h_{f_2}^{l_1} & \cdots & h_{f_2}^{l_{3s-1}} \\ l_0 \cdot h_{f_2}^{l_0} & l_1 \cdot h_{f_2}^{l_1} & \cdots & l_{3s-1} \cdot h_{f_2}^{l_{3s-1}} \\ l_0^2 \cdot h_{f_2}^{l_0} & l_1^2 \cdot h_{f_2}^{l_1} & \cdots & l_{3s-1}^2 \cdot h_{f_2}^{l_{3s-1}} \\ \vdots & \vdots & \ddots & \vdots \\ h_{f_s}^{l_0} & h_{f_s}^{l_1} & \cdots & h_{f_s}^{l_{3s-1}} \\ l_0 \cdot h_{f_s}^{l_0} & l_1 \cdot h_{f_s}^{l_1} & \cdots & l_{3s-1} \cdot h_{f_s}^{l_{3s-1}} \\ l_0^2 \cdot h_{f_s}^{l_0} & l_1^2 \cdot h_{f_s}^{l_1} & \cdots & l_{3s-1}^2 \cdot h_{f_s}^{l_{3s-1}} \end{bmatrix}, \tag{3.63}$$

is full rank with probability 1 if the frequencies $f_1, \cdots, f_j, \cdots, f_s$ *are drawn uniformly at random in* $[0, 1]$.

Proof of Proposition 3.7. We show $\boldsymbol{A}_s$ is full-rank by proving that its determinant, $det(\boldsymbol{A}_s) = |\boldsymbol{A}_s|$ is a non-zero polynomial. For $s = 1$, we have the matrix,

$$\boldsymbol{A}_1 = \begin{bmatrix} h_{f_1}^{l_0} & h_{f_1}^{l_1} & h_{f_1}^{l_2} \\ l_0 \cdot h_{f_1}^{l_0} & l_1 \cdot h_{f_1}^{l_1} & l_2 \cdot h_{f_1}^{l_2} \\ l_0^2 \cdot h_{f_1}^{l_0} & l_1^2 \cdot h_{f_1}^{l_1} & l_2^2 \cdot h_{f_1}^{l_2} \end{bmatrix}. \tag{3.64}$$

We note that $|\boldsymbol{A}_s|$ easily reduces to a Vandermonde determinant (of order 3), so that $|\boldsymbol{A}_s| = (l_2 - l_1)(l_2 - l_0)(l_1 - l_0)h_{f_1}^{l_0+l_1+l_2}$, which is a non-zero polynomial because l_0, l_1,

and l_2 are distinct sample indices.

Let us now assume that, for $s > 1$, $|\boldsymbol{A}_s|$ is a non-zero polynomial. We would like to show that $|\boldsymbol{A}_{s+1}|$ is also a non-zero polynomial. We have,

$$\boldsymbol{A}_{s+1} = \begin{bmatrix} \boldsymbol{A}_s & \boldsymbol{B} \\ \boldsymbol{C} & \boldsymbol{D} \end{bmatrix}, \tag{3.65}$$

where

$$\boldsymbol{D} = \begin{bmatrix} h_{f_{s+1}}^{l_{3s}} & h_{f_{s+1}}^{l_{3s+1}} & h_{f_{s+1}}^{l_{3s+2}} \\ l_{3s}h_{f_{s+1}}^{l_{3s}} & l_{3s+1}h_{f_{s+1}}^{l_{3s+1}} & l_{3s+2}h_{f_{s+1}}^{l_{3s+2}} \\ l_{3s}^2h_{f_{s+1}}^{l_{3s}} & l_{3s+1}^2h_{f_{s+1}}^{l_{3s+1}} & l_{3s+2}^2h_{f_{s+1}}^{l_{3s+2}} \end{bmatrix}_{3\times 3}. \tag{3.66}$$

Noting that the determinant of row echelon form is same as the original matrix, we obtain the row echelon form (REF) for the matrix $\boldsymbol{D}$ as follows:

$$REF(\boldsymbol{D}) = \begin{bmatrix} h_{f_{s+1}}^{l_{3s}} & h_{f_{s+1}}^{l_{3s+1}} & h_{f_{s+1}}^{l_{3s+2}} \\ 0 & (l_{3s+1}-l_{3s})h_{f_{s+1}}^{l_{3s+1}} & (l_{3s+2}-l_{3s})h_{f_{s+1}}^{l_{3s+2}} \\ 0 & 0 & (l_{3s+2}-l_{3s})(l_{3s+2}-l_{3s+1})h_{f_{s+1}}^{l_{3s+2}} \end{bmatrix}. \tag{3.67}$$

Let $a_{i,j}$ be the element of the matrix $\boldsymbol{A}_{s+1}$ in ith row and jth column, then by the Leibniz formula for determinants,

$$\begin{aligned} |\boldsymbol{A}_{s+1}| &= \sum_{\sigma\in S_{3s+3}} sgn(\sigma)a_{1,\sigma(1)}a_{2,\sigma(2)}\cdots a_{3s+3,\sigma(3s+3)} \\ &= P_{l_{3s}+l_{3s+1}+l_{3s+2}}(h_{f_{s+1}})|\boldsymbol{A}_s| + P(h_{f_1},\cdots,h_{f_s},h_{f_{s+1}}), \end{aligned} \tag{3.68}$$

where sgn is the sign function of permutations in the permutation group S_{s+1}, $P(h_{f_1},\cdots,h_{f_s},h_{f_{s+1}})$ is a multivariate polynomial, and $P_{l_{3s}+l_{3s+1}+l_{3s+2}}(h_{f_{s+1}})$ is a non-zero univariate monomial in $h_{f_{s+1}}$ of degree $l_{3s}+l_{3s+1}+l_{3s+2}$. From the row echelon form in (3.67), we recognize that the highest degree of the variable $h_{f_{s+1}}$ in the expansion

(3.68) is $l_{3s} + l_{3s+1} + l_{3s+2}$. Note that the polynomial $P(h_{f_1}, \cdots, h_{f_s}, h_{f_{s+1}})$ has lower degree in $h_{f_{s+1}}$ than $P_{l_{3s}+l_{3s+1}+l_{3s+2}}(h_{f_{s+1}})$. Since $|\boldsymbol{A}_s|$ is a non-zero polynomial, the coefficient of $h_{f_{s+1}}^{l_{3s}+l_{3s+1}+l_{3s+2}}$ is also a non-zero polynomial. Therefore, $|\boldsymbol{A}_{s+1}|$ is a non-zero polynomial. Further, the probability that one randomly picks the frequencies over $[0, 1]$ such that each h_{f_j} is a root of this non-zero polynomial is zero.[2] Thus, by induction, $|\boldsymbol{A}_s|$ is non-zero with probability 1. □

The formulation in (3.38) generalizes the prior information. As the cases of *block priors* and *known poles* indicate, the more we know about the spectral structure of the signal, precise formulations of atomic norm minimization can be evaluated to boost signal recovery. If all poles are known in the sense of *known poles* algorithm (i.e., the amplitudes and phases of all known poles are unknown), then the signal x can be uniquely reconstructed using the randomly sampled support $x_{\mathcal{M}}$ where $|\mathcal{M}| = s$. Further, it is well known that if the signal is uniformly sampled, then the Prony's method can uniquely reconstruct the signal x using no more than $2s$ samples. In comparison, our results from Theorem 3.6 show that if each of the poles are *approximately known*, then the unique reconstruction of the signal x requires no more than $3s$ samples. A comparison of Theorem VI.3 with the result in [168, Theorem I.1] indicates that when every signal frequency is known within a sufficiently small frequency subband, the signal can be recovered with lesser samples and without the restriction of minimum frequency separation.

[2] An analogous argument for a polynomial with roots over a finite field can be found in Schwartz-Zippel-DeMillo-Lipton lemma [61, 158, 201].

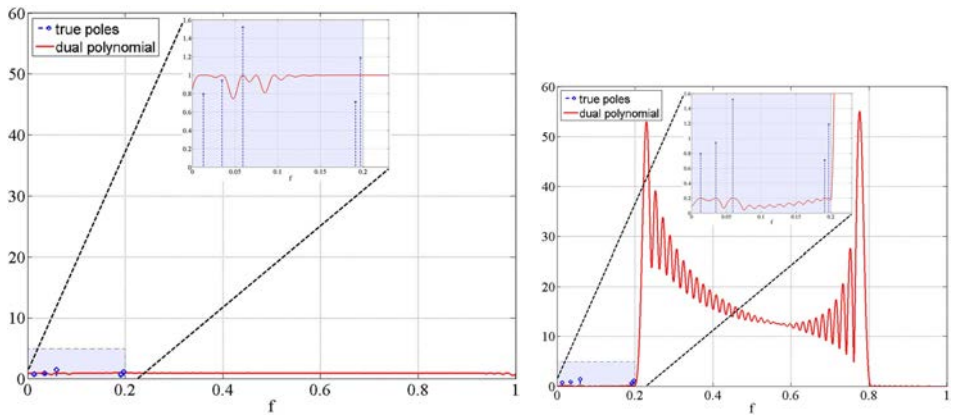
3.1.7 Numerical experiments

We evaluated our algorithms through numerical experiments using the SDPT3 [178] solver for the semidefinite programs. In all experiments, for a particular realization of the signal, the phases of the signal frequencies were sampled uniformly at random in $[0, 2\pi)$. The amplitudes $|c_j|, j = 1, \cdots, s$ were drawn randomly from the distribution $0.5 + \chi_1^2$ where χ_1^2 represents the chi-squared distribution with 1 degree of freedom.

3.1.7.1 Probabilistic priors

We evaluated the semidefinite program (3.38) for the case when $p = 2$. Here, $\mathcal{B}_1 = [0, 0.2]$ and $\mathcal{B}_2 = (0.2, 1]$ so that $\mathcal{B}_1 \bigcup \mathcal{B}_2 = [0, 1]$. We consider the situation when the probability of occurrence of signal frequency in $\mathcal{B}_1$ is 1000 times higher than $\mathcal{B}_2$. This results in the pdf values of $p_F(f)|_{\mathcal{B}_1} = 4.9801$ and $p_F(f)|_{\mathcal{B}_2} = 0.005$. A suitable sub-optimal choice of $w(f)$ could be simply $w(f) = 1/p_F(f)$, so that the associated weights are given by $w_1 = 0.2008$ and $w_2 = 200.8000$. For each random realization of the signal, the signal frequencies are drawn randomly based on the given probability density function.

Experiment A.1. A simple illustration of the signal recovery using (3.38) is shown through frequency localization in Figure 3.4. For a signal of dimension $n = 64$ and number of frequencies $s = 5$, Figure 3.4a shows that even when all samples are observed ($m = 64$), the standard atomic norm minimization (3.8) is unable to recover any of the frequencies, for the maximum modulus of the dual polynomial assumes a value of



(a) Without any priors (b) With probabilistic priors

Figure 3.4: Frequency localization using dual polynomial for $\{n, s, m\} = \{64, 5, 64\}$. The probabilistic priors are $p_F(f)|_{\mathcal{B}_1=[0,0.2]} = 4.9801$ and $p_F(f)|_{\mathcal{B}_2=(0.2,1]} = 0.005$. The insets show the same plot on a smaller scale.

unity at many other frequencies. However, given the probabilistic priors, semidefinite program (3.38) is able to perfectly recover all the frequencies as shown in Figure 3.4b. Here, $|Q(f_j)| = w_1 = 0.2008$ for $f_j \in \mathcal{F} \subset \mathcal{B}_1$, and $|Q(f_j)| = w_2 = 200.8$ for $f_j \in \mathcal{F} \subset \mathcal{B}_2$.

Experiment A.2. A comparison of the statistical performance of (3.38) with the standard atomic norm for $n = 64$ is shown in Figure 3.5 over 1000 trials. Here, the pdf $p_F(f)$ is 1000 times higher in the subband $(0.3, 0.7]$ than the rest of the spectrum. We note that the weighted atomic norm is about twice more successful than the standard atomic norm in recovering the signal frequencies.

3.1.7.2 Block priors

We evaluated the performance of spectrum estimation with block priors through numerical simulations for the semidefinite program in (3.42). Here, for every random realization of the signal, the frequencies are drawn uniformly at random in the set of

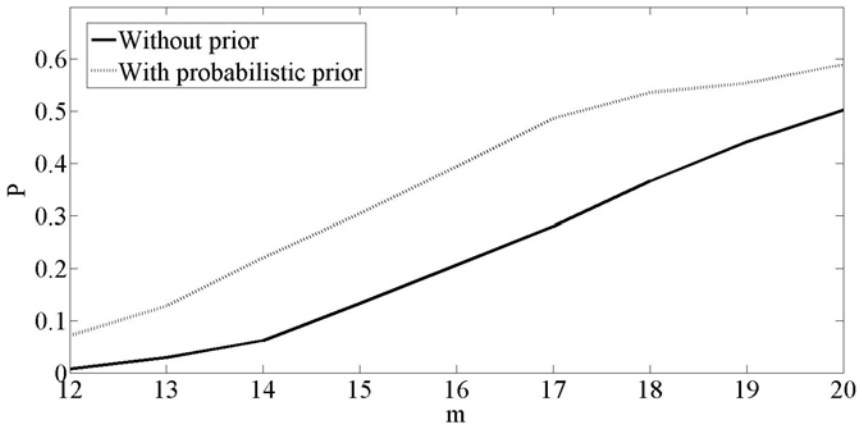
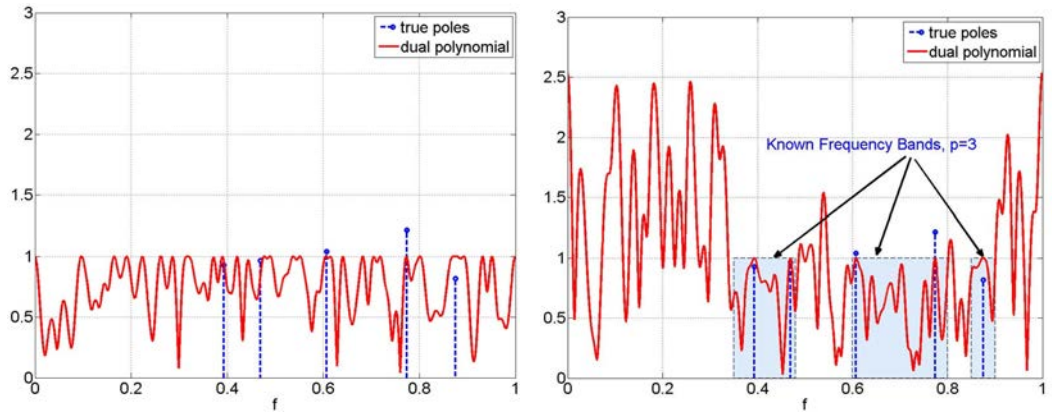


Figure 3.5: The probability P of perfect recovery over 1000 trials for $\{n, s\} = \{64, 5\}$. The probabilistic priors are $p_F(f)|_{\mathcal{B}_1=\{[0,0.3]\cup(0.7,1]\}} = 0.0025$ and $p_F(f)|_{\mathcal{B}_2=(0.3,0.7]} = 2.4963$.

subbands $\mathcal{B} = \bigcup_{k=1}^{p} \mathcal{B}_k \subset [0, 1]$.

Experiment B.1. We first illustrate our approach through an example in Figure 3.6. Here for $n = 64$, we drew $s = 5$ frequencies uniformly at random within $p = 3$ subbands in the domain $[0, 1]$ without imposing any minimum separation condition. Here, $\mathcal{B} = (0.35, 0.48) \cup (0.60, 0.80) \cup (0.85, 0.90)$. A total of $m = 20$ observations were randomly chosen from n regular time samples to form the sample set $\mathcal{M}$. In the absence of any prior information, we solve (3.11) and show the result of frequency localization in Figure 3.6a. Here, it is difficult to pick a unique set of $s = 5$ poles for which the maximum modulus of the dual polynomial is unity (which will actually correspond to recovered frequency poles). On the other hand, when block priors are given, Figure 3.6b shows that solving (3.42) provides perfect recovery of all the frequency components, where the recovered frequencies correspond to unit-modulus points of the dual polynomial.

Experiment B.2. We then give a statistical performance evaluation of our new

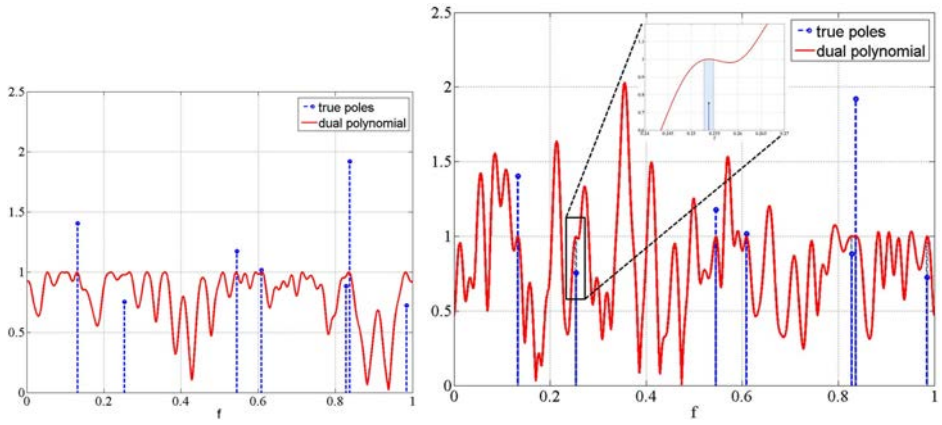


(a) Without any priors (b) With block priors

Figure 3.6: Frequency localization using dual polynomial for $\{n, s, m\} = \{64, 5, 20\}$. The block priors are $\mathcal{B} = [0.35, 0.48] \cup [0.60, 0.80] \cup [0.85, 0.90]$.

method, compared with atomic norm minimization without any priors (3.11). The experimental setup and block priors are the same as in Figure 3.6 and no minimum separation condition was assumed while drawing frequencies uniformly at random in the set $\mathcal{B}$. Figure 3.8 shows the probability P of perfect recovery for the two methods for fixed $n = 64$ but varying values of m and s. For every value of the pair $\{s, m\}$, we simulate 100 trials to compute P. We note that if the frequencies are approximately known, our method greatly enhances the recovery of continuous-valued frequencies.

Experiment B.3. To illustrate the theoretical result of Theorem 3.6, we now consider the block prior problem when each of the frequencies are known to lie in extremely small subintervals. For the triplet $\{n, s, m\} = \{64, 7, 18\}$, Figure 3.7 depicts the frequency localization. In the absence of any prior knowledge, the standard atomic norm minimization of (3.8) fails to locate any of the signal frequencies (Figure 3.7a). However, if the frequencies are approximately known (or, in



(a) Without any priors (b) With block priors

Figure 3.7: Frequency localization using dual polynomial for $\{n, s, m\} = \{64, 7, 18\}$. The block priors consist of small frequency bands around each true pole f_j such that $\mathcal{B} = \cup_{k=1}^{s} \mathcal{B}_k = \cup_{k=1}^{s} [f_j - 0.001, f_j + 0.001]$. The bottom plot has been magnified in the inset to show the size of the block prior.

other words, the frequency subband of the block prior is very small), then the semidefinite program in (3.42) perfectly recovers the signal requiring not more than $3s$ number of samples ($m = 18 < 21 = 3s$), as shown in Figure 3.7b. Here, the block priors consist of narrow frequency bands around each true pole f_j such that $\mathcal{B} = \cup_{k=1}^{s} \mathcal{B}_k = \cup_{k=1}^{s} [f_j - 0.001, f_j + 0.001]$.

Experiment B.4. For the same signal dimension, size and number of blocks as in the previous experiment, Figure 3.9 shows a comparison of statistical performance of block prior method with the standard atomic norm minimization over 100 trials. For every s, the parameter m was varied until m was at least $3s$. (Note that the perfect recovery with 3s samples in Theorem 3.6 holds only when the block prior is arbitrarily small.)

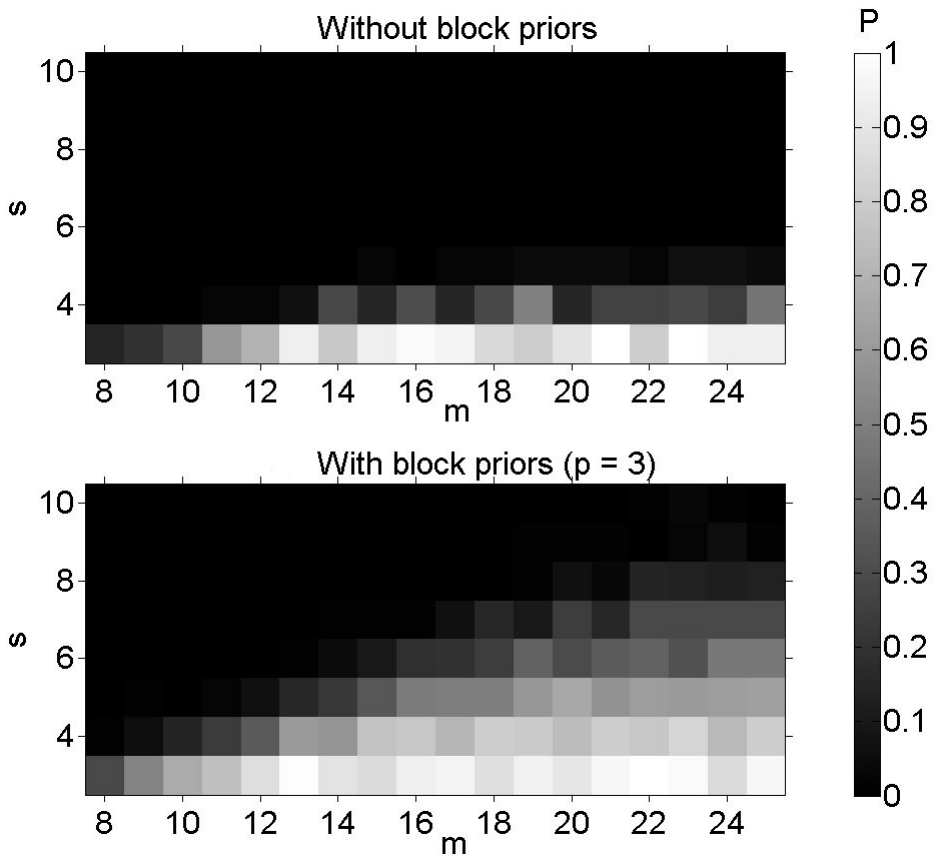


Figure 3.8: The probability P of perfect recovery over 100 trials for $n = 64$. The block priors are same as in Figure 3.6.

3.1.7.3 Known poles

We evaluated the *known poles* algorithm through a number of simulations to solve the semidefinite program (3.45). In all our experiments, the s frequencies of the artificially generated signal were drawn at random in the band $[0, 1]$. Except for Experiment 4, the sampled frequencies were also constrained to have the minimum modulo spacing of $\Delta f = 1/\lfloor (n-1)/4 \rfloor$ between the adjacent frequencies. This is the theoretical resolution condition for the results in [168], although numerical experiments suggested that frequencies could be closer, i.e., Δf could be $1/(n-1)$. While working

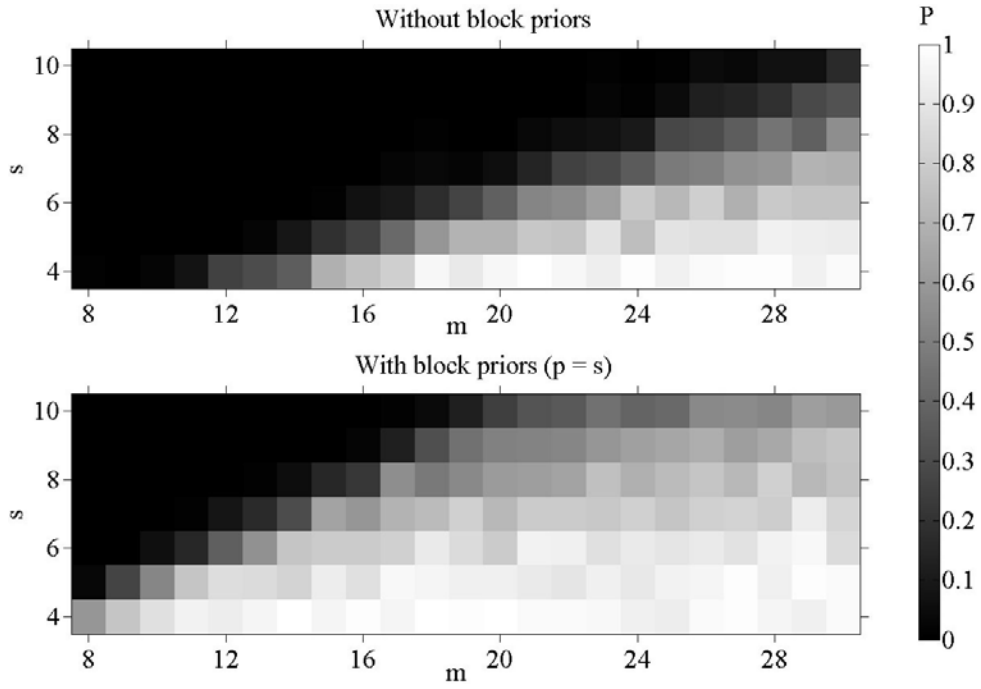


Figure 3.9: The probability P of perfect recovery over 100 trials for $n = 64$. The block priors consist of small frequency bands around each true pole f_j as in Figure 3.7.

with the *known poles*, we draw the first known frequency uniformly at random from the set of s frequencies. As the number p of *known poles* increases, we retain the previously drawn known frequencies and draw the next known frequency uniformly at random from the remaining set of existing signal frequencies.

Experiment C.1. We simulated a low-dimensional model with the triple $(n, m, s) = (32, 9, 4)$ and first solved the semidefinite program (3.8) which does not use any prior information, i.e., $p = 0$. For the same realization of the signal, we then successively increase p up to $s-1$, and solve the optimization (3.45) of the *known poles* algorithm. At every instance of solving an SDP, we record the number k of successfully recovered frequencies along with their complex coefficients. This number also includes

the known frequencies if the recovery process returns exact values of their complex coefficients. $k = s$ corresponds to *complete success*, i.e., recovering all of the unknown spectral content. $k = 0$ is *complete failure*, including the case when the complex coefficients of the known frequencies could not be recovered. Figure 3.10 shows the probability P of recovering k frequencies over 1000 trials. Although the complex coefficients of the known frequencies were unknown, the *known poles* algorithm increases the probability of accurately recovering all or some of the unknown spectral content.

Experiment C.2. We repeat the first experiment for the higher-dimensional pair $(n, m) = (256, 40)$ and vary s. The probability P over 100 random realizations of the signal is shown in Figure 3.11 for selected values of s. We observe that the probability of successfully recovering all the frequencies using the *known poles* Algorithm 3.2 increases with p.

Experiment C.3. Figure 3.12 shows the probability P of *complete success* as a function of m over 100 trials for the twin $(n, s) = (80, 6)$. We note that the *known poles* algorithm achieves the same recovery probability when compared to (3.8) with a smaller number of random observations.

Experiment C.4. We now consider these two cases: (a) when $\Delta f = 1/(n-1)$, the resolution limit for the numerical experiments in [168], and (b) when the frequencies are drawn uniformly at random and do not adhere to any minimum resolution conditions. Figure 3.13 shows the probability P of recovering k frequencies over 1000 trials for the triple $(n, m, s) = (40, 15, 7)$. We note that the probability of *complete success* with *known poles* suffers relatively little degradation for the random frequency resolutions.

These trials include instances when the minimum resolution condition does not hold, formulation in (3.8) shows *complete failure* but the *known poles* algorithm recovers the unknown spectral content with *complete success*.

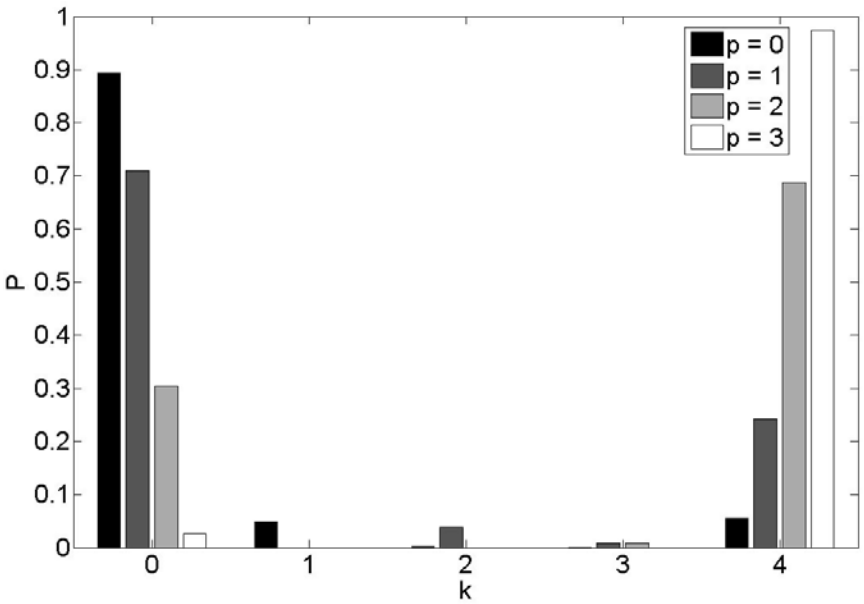


Figure 3.10: The probability P of recovering the unknown spectral content. The probability is computed for 1000 random realizations of the signal for the triple $(n, m, s) = (32, 9, 4)$. (For $k > 0$, $k \leq p$ being the invalid cases, the corresponding bars have been omitted.)

3.1.8 Discussion

Our application of properties of positive trigonometric polynomials establishes a general framework to employ prior information in spectral estimation of undersampled signals. Further interesting investigations are now possible in this framework, most notably determination of theoretical limit of super-resolution and recovery of signals in presence of additive noise. We now provide brief insights into these aspects

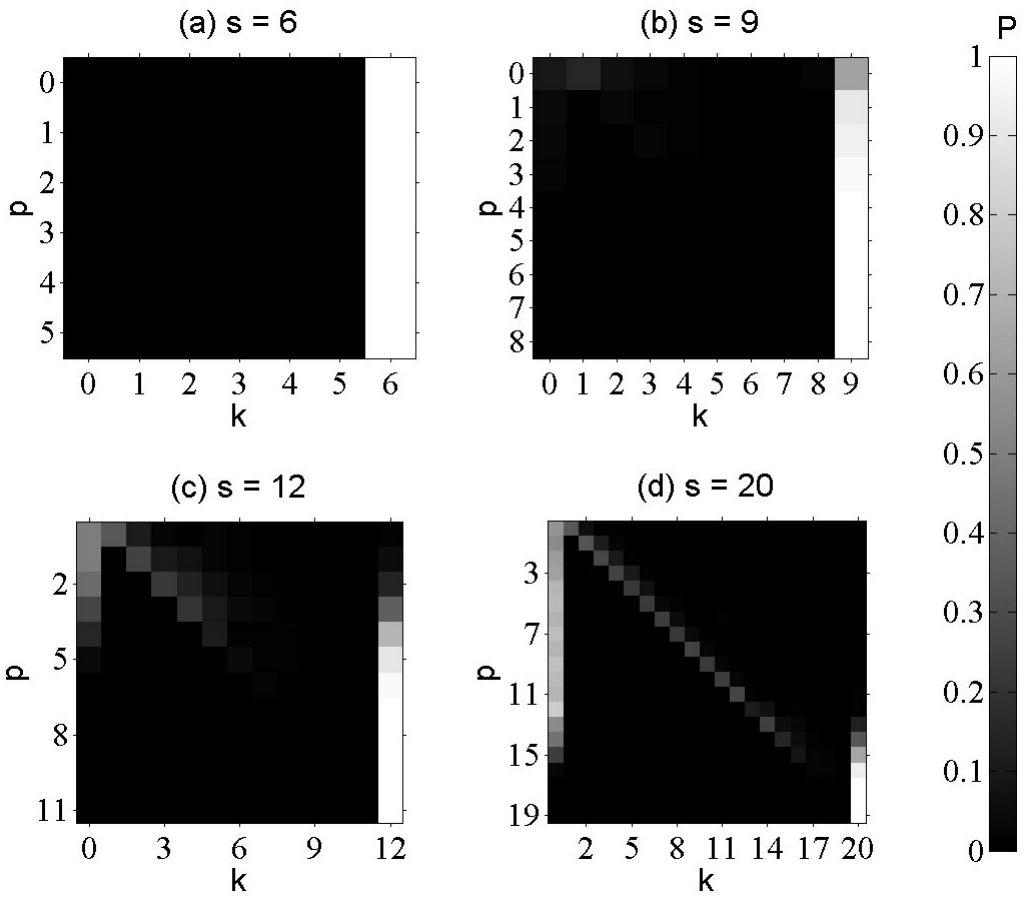


Figure 3.11: The probability P of recovering the unknown spectral content for selected values of s. The probability is computed for 100 random realizations of the signal with $(n, m) = (256, 40)$. (The lower diagonal cases when $k > 0$, $k \leq p$ are invalid, and do not contribute to the result.)

concerning the performance of our algorithms.

3.1.8.1 Super-resolution

Our numerical experiments demonstrate that prior information aids in improving the classical super-resolution limit of $\frac{1}{n}$. For example, the standard super-resolution limit for the signal in Figure 3.4 is $\frac{1}{64} = 0.0156$, yet our approach using weighted atomic norm could resolve the frequency pair $\{0.1908, 0.1966\}$ with an inter-

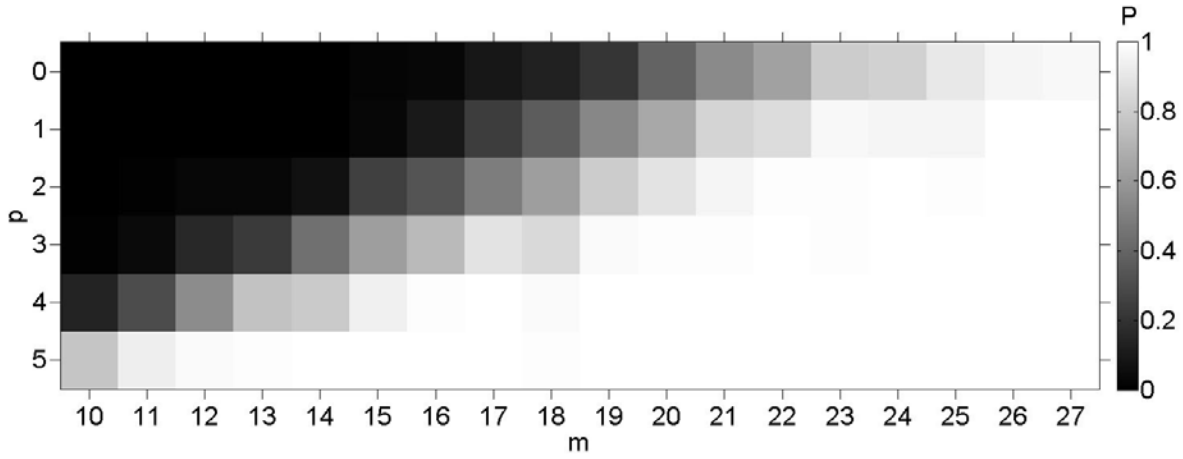


Figure 3.12: A higher probability P of recovering all the unknown frequency content can be achieved with a smaller number m of random observations using the *known poles* algorithm. The probability is computed for 100 random realizations with $(n, s) = (80, 6)$.

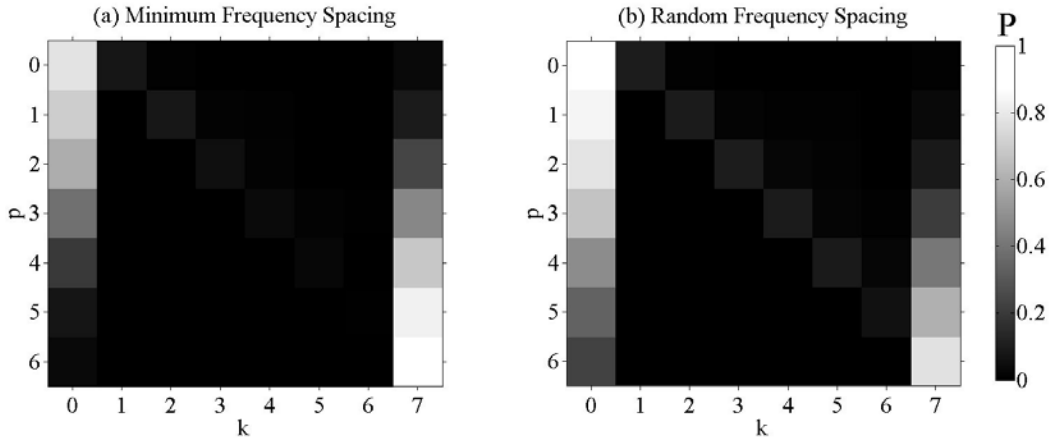


Figure 3.13: Performance of the *known poles* algorithm when the frequencies do not satisfy any nominal resolution conditions. The probability P of successfully recovering k frequencies is computed for 1000 realizations of the signal with dimensions $(n, m, s) = (40, 15, 7)$. (a) $\Delta f = 1/(n-1)$ (b) Frequencies are selected uniformly at random in the band $[0, 1]$.

frequency separation of 0.0058. Further, statistical comparisons in Figures 3.5 and 3.8, where the number of frequencies s are gradually increased for a fixed signal dimension n so that the super-resolution limit of $\frac{1}{n}$ is violated more often than not, show

superior recovery performance when prior information algorithms are applied. The clearest instance of lower super-resolution limit in the presence of prior information appears in the Figure 3.13 where the probability of signal recovery with *known-poles* algorithm is higher even when the frequencies are generated without the minimum separation condition.

As one of the future directions, we intend to pursue the theoretical limits of super-resolution in presence of prior information. We note that the super-resolution limit with priors must be at least same as the limit derived in [27] and [168]. For example, the set $\mathcal{A}_\alpha$ of atoms in our formulation of block priors is a subset of the set $\mathcal{A}_\beta$ of atoms in [27] and [168]. If the methods described in [27] and [168] are able to correctly recover the set $\mathcal{A}_\gamma$ of original frequencies ($\mathcal{A}_\gamma \subseteq \mathcal{A}_\alpha$), then the set $\mathcal{A}_\gamma$ of atoms achieves the smallest atomic norm among the atoms in the set $\mathcal{A}_\beta$. Since $\mathcal{A}_\gamma \subseteq \mathcal{A}_\alpha \subseteq \mathcal{A}_\beta$, the atoms in $\mathcal{A}_\gamma$ also achieve the smallest atomic norm within the set $\mathcal{A}_\alpha$. Therefore, at the very least, our method can achieve a super-resolution limit of [27] and [168], that is, $\frac{4}{n}$.

3.1.8.2 Noisy signal with priors

In real applications, the spectral estimation has an additional challenge of recovering signals in presence of noise. Recent research suggests spectral super-resolution with noise [26, 140] remains a rich problem, that we would not able to address in detail within the context of this paper. However, our methods can be developed further to recover a spectrally sparse signal in presence of noise. Here, we

first derive the dual problem in presence of noise and prior information, and then provide a result of numerical experiment as an evidence of improvement and stability of performance.

Suppose that the observations $x'_{\mathcal{M}}$ satisfy $\|x'_{\mathcal{M}} - x_{\mathcal{M}}\|_2^2 \leq \delta$, where δ is a constant. In this case, the primal atomic norm minimization problem would be

$$\begin{aligned} \underset{\hat{x}}{\text{minimize}} \quad & \|\hat{x}\|_{\mathcal{A}} \\ \text{subject to} \quad & \|\hat{x}_{\mathcal{M}} - x'_{\mathcal{M}}\|_2^2 \leq \delta, \end{aligned} \tag{3.69}$$

By the Lagrangian duality, we derive the corresponding dual problem as

$$\begin{aligned} \underset{q}{\text{maximize}} \quad & \langle q_{\mathcal{M}}, x'_{\mathcal{M}} \rangle_{\mathbb{R}} - \sqrt{\delta}\|q_{\mathcal{M}}\|_2 \\ \text{subject to} \quad & \|q\|_{\mathcal{A}}^* \leq 1 \\ & q_{\mathcal{N} \setminus \mathcal{M}} = 0. \end{aligned} \tag{3.70}$$

Note that when δ is 0, the optimization problems 3.69 and 3.70 are equivalent to the primal (3.3) and dual (3.11) problems of atomic norm minimization with noiseless observations. The semidefinite program for the primal problem (3.69) would be identical to (3.8) except that the equality constraint in (3.8) would now be replaced by $\|\hat{x}_{\mathcal{M}} - x'_{\mathcal{M}}\|_2^2 \leq \delta$. As for the dual problem, we note that only the objective function has been altered in presence of noise while the constraints are unchanged. On the other hand, our prior information algorithms change the constraints of the dual problem to develop the corresponding semidefinite program. Therefore, to recover the signal in presence of noise, one only needs to add the term $-\sqrt{\delta}\|q_{\mathcal{M}}\|_2$ to the objective functions of prior information formulations such as (3.38) and (3.42) in order to develop

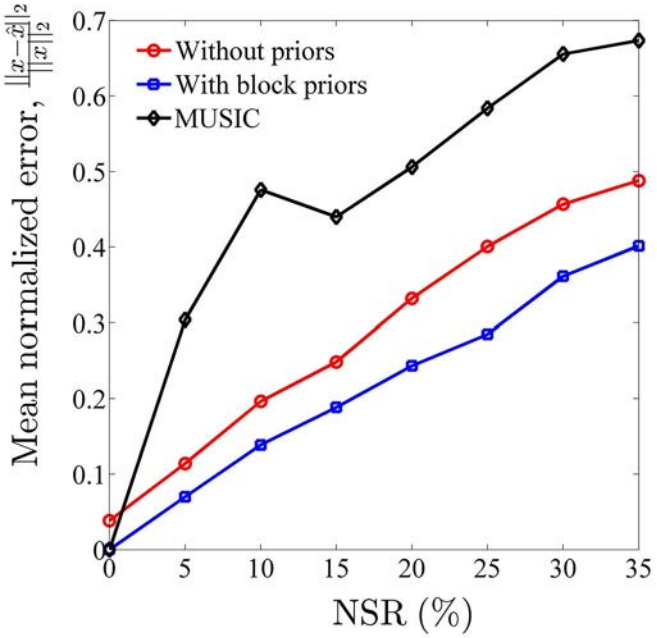


Figure 3.14: **M**ean normalized errors in the recovered signal as a function of Noise-to-Signal Ratio (NSR) for $n = 30$, $s = 4$, $m = 15$, and $\mathcal{B} = (0.35, 0.40) \cup (0.60, 0.70) \cup (0.85, 0.90)$.

the corresponding SDP.

As an illustration of this approach, we evaluated our block prior SDP in presence of noise through numerical experiments. We didn't restrict the frequencies to any minimum separation and drew them uniformly at random in these experiments. The complex coefficients were drawn randomly exactly as explained in the beginning of Section 3.1.7. We didn't change the values of the following parameters for any of the signal realizations: signal dimension $n = 30$, number of frequencies $s = 4$, number of random observations $m = 15$, and block priors $\mathcal{B} = (0.35, 0.40) \cup (0.60, 0.70) \cup (0.85, 0.90)$. We then added complex i.i.d. Gaussian noise to m random samples $x_{\mathcal{M}}$ to obtain the noisy observations $x'_{\mathcal{M}}$. The noise was scaled such that its energy δ provides the specified Noise-to-Signal Ratio, $NSR = \frac{\sqrt{\delta}}{||x||_2}$. We evaluated the recovery

performance in terms of the normalized error $\frac{\|x-\hat{x}\|_2}{\|x\|_2}$.

We then compared the mean normalized errors over 100 realizations for different values of NSR with and without block priors. Figure 3.14 shows the mean normalized errors for cases when the signal was successfully recovered by either of the two methods - with or without priors - in the absence of noise. Here, the noiseless scenario corresponds to $NSR = 0$. We note that the prior information formulation yields a higher stability in presence of noise. Similar performance was observed when we included those realizations where the noiseless recovery was not successful.

Figure 3.14 also shows the performance of a classical noise subspace method, namely MUSIC [157]. We take the first 1 to m samples of the signal for the numerical experiments of MUSIC, as this method admits only uniformly-sampled signals. Figure 3.14 shows that the signal recovery through MUSIC has larger errors than our method in the presence of noise. Since the minimum frequency separation constraints do not apply to MUSIC, it performs better than the standard atomic norm minimization when $NSR = 0$. On the other hand, our prior information techniques achieve super-resolution limits closer to MUSIC in the noiseless case. One could also choose the equi-spaced m samples for MUSIC by uniformly sampling m points over the entire n-dimensional signal vector. In this case, the frequencies f_1 and $f_2 = 0.5 + f_1$, where $\{f_1, f_2\} \in [0, 1]$, cannot be distinguished by MUSIC, leading to even higher errors than shown in Figure 3.14 .

3.2 Block Iterative Reweighted Algorithms

3.2.1 Introduction

Compressed sensing promises to perform signal recovery using a smaller number of samples than required by the Nyquist-Shannon sampling theorem. In the compressed sensing framework, a sparse signal x is recovered from the observation vector y even though the dimension of y is much smaller than the dimension of x. Since compressed sensing reduces the sampling rate in recovering sparse signals, it has made great impacts in various signal processing areas [77].

Compressed sensing has also found application to the problem of *line spectral estimation*, which aims to estimate spectral information from few observations. Early-stage compressed sensing frameworks for spectral estimation [49, 71] assumed that the frequencies of spectrally sparse signals were located on discretized grid points in the frequency domain. However, in practice, frequencies can take values in a continuous domain, giving rise to the so-called *basis mismatch* problem [49] when the discretization of the frequency domain is not fine enough.

The breakthrough theory of *super-resolution* [27] proposed by Candès and Fernandez-Granda states that sparse continuous-valued frequencies can be exactly recovered through total variation minimization using a set of n *uniformly* spaced time samples, provided the minimum separation between any two frequencies is $4/n$. In order to recover continuous-valued frequencies from few *randomly* chosen nonuniformly-spaced time samples, Tang *et al.* proposed off-the-grid compressed sensing that employs atomic norm minimization for frequency recovery [168]. Later, it was shown

that the ℓ_1 minimization over the fine discrete dictionary provides an approximate solution to the atomic norm minimization [167].

In this paper, we are interested in recovering spectrally sparse signals with as few random time samples as possible. It is then natural to ask whether there are efficient frequency recovery algorithms that can further improve the performance or relax the frequency separation conditions when compared with the total variation minimization or atomic norm minimization. We propose new iterative algorithms to enhance the performance of recovering continuous-valued frequency. In our iterative algorithms, we estimate the frequency support information from previous iterations, and use the support information as *block prior* [137] for reweighted atomic norm minimization in later iterations. Numerical results show that we can improve recovery performance by exploiting the block prior provided in earlier iterations.

We remark that there are quite a few works in the literature [32, 40, 78, 142, 184, 194] where iterative reweighted methods have been used to improve sparse recovery performance in compressed sensing. However, the sparse signal recovery is considered over a finite discrete dictionary in [32, 40, 78, 142, 184]. Besides [194], only our work considers recovering continuous-valued frequencies by directly reweighting in the continuous dictionary through a semi-definite program (SDP). Our work differs from [194] in that we provide different reweighting schemes that lead to improved signal recovery performance. In [194], the authors set the reweighting weight $w(f)$ for a frequency $f \in [0, 1]$ according to correlations between frequency atoms (see e.g. Theorem 3 of [194]). In contrast, our method allows $w(f)$ to take more gen-

eral forms through the dual program of weighted atomic minimization under general weights [138], thereby lending more flexibility to incorporating external prior information and prior information passed on from earlier algorithm iterations. Numerical experiments show that our iterative algorithms improve both the recovery performance and the execution time, compared with [168] and [194].

3.2.2 Background on standard and weighted atomic norm minimization algorithms

In this paper, we denote the set of complex numbers, real numbers, positive integers and natural numbers including 0 as $\mathbb{C}$, $\mathbb{R}$, $\mathbb{Z}^+$, and $\mathbb{N}$ respectively. We reserve calligraphic uppercase letters for index sets. When we use an index set $\mathcal{K}$ as the subscript of a vector x or a matrix F, i.e., $x_{\mathcal{K}}$ or $F_{\mathcal{K}}$, it represents the part of the vector x over index set $\mathcal{K}$ or the columns of the matrix F over index set $\mathcal{K}$ respectively.

Let $x^\star$ be a spectrally sparse signal expressed as a sum of k complex exponentials as follows:

$$x_l^\star = \sum_{j=1}^{k} c_j^\star e^{i2\pi f_j^\star l} = \sum_{j=1}^{k} |c_j^\star| a(f_j^\star, \phi_j^\star)_l, \quad l \in \mathcal{N}, \tag{3.71}$$

where $f_j^\star \in [0,1]$ represents a frequency, $c_j^\star = |c_j^\star| e^{i\phi_j^\star}$ is its coefficient, and $\phi_j^\star \in [0, 2\pi]$ is its phase, $\mathcal{N} = \{l \,:\, 0 \le l \le n-1, l \in \mathbb{N}\}$ is the set of time indices. Here, $a(f_j^\star, \phi_j^\star) \in \mathbb{C}^{|\mathcal{N}|}$ is a *frequency-atom*, with the l-th element given by $a(f_j^\star, \phi_j^\star)_l = e^{i(2\pi f_j^\star l + \phi_j^\star)}$. In particular, when phase is 0, we denote the frequency-atom simply as $a(f_j)$. We assume that the signal in (3.71) is observed over the time index set $\mathcal{M} \subseteq \mathcal{N}$, $|\mathcal{M}| = m \le n$, where m observations are chosen randomly. Our goal is to recover all the frequencies with the smallest possible number of observations. Estimating frequencies is not trivial

because they are in continuous domain, and their phases and magnitudes are also unknown.

The atomic norm of a signal x and its dual norm [168, Eq. (II.7)] are defined respectively as follows:

$$\|x\|_{\mathcal{A}} = \inf\{\sum_j |c_j| : x = \sum_j c_j a(f_j)\}, \tag{3.72}$$

$$\|q\|_{\mathcal{A}}^* = \sup_{\|x\|_{\mathcal{A}} \le 1} \langle q, x\rangle_{\mathbb{R}} = \sup_{\substack{\phi\in[0,2\pi],\\ f\in[0,1]}} \langle q, e^{i\phi}a(f,0)\rangle_{\mathbb{R}} = \sup_{f\in[0,1]} |\langle q, a(f)\rangle|, \tag{3.73}$$

where $\langle q, x\rangle_{\mathbb{R}}$ represents the real part of the inner product $x^H q$. Here, the superscript H is used for the conjugate transpose. In [168], the authors proposed the following atomic norm minimization to recover a spectrally sparse signal $x^\star$ using randomly chosen time samples $\mathcal{M} \subseteq \mathcal{N}$:

$$\begin{aligned} &\underset{x}{\text{minimize}} \ \ \|x\|_{\mathcal{A}} \\ &\text{subject to} \ \ x_j = x_j^\star, \ \ j \in \mathcal{M}. \end{aligned} \tag{3.74}$$

The dual problem of (3.74) is

$$\begin{aligned} &\underset{q}{\text{maximize}} \ \ \langle q_{\mathcal{M}}, x_{\mathcal{M}}^\star\rangle_{\mathbb{R}} \\ &\text{subject to} \ \ q_{\mathcal{N}\setminus\mathcal{M}} = 0, \ \|q\|_{\mathcal{A}}^* \le 1. \end{aligned} \tag{3.75}$$

The constraint $\|q\|_{\mathcal{A}}^* \le 1$ in (3.75) can be changed to $\sup_{f\in[0,1]} |\langle q, a(f)\rangle| \le 1$ using (3.73). We label $\langle q, a(f)\rangle$ as the *dual polynomial* $Q(f)$. Since the Slater condition is satisfied in (3.74), there is no duality gap between (3.74) and (3.75) [20]. Moreover, the estimated spectral content comprises the frequencies at which the absolute value of the dual polynomial, which is derived from q (obtained as a solution of (3.75)),

attains the maximum modulus of unity. We refer the reader to [168] for details. The off-the-grid compressed sensing approach in [168] demonstrated that with randomly chosen observation data, one can correctly obtain frequency information by solving the atomic norm minimization. However, the atomic norm minimization requires a certain minimum separation between frequencies for successful recovery.

In [137], we considered frequency recovery with external prior information, and showed that if the frequencies are known to lie in frequency subbands, we can obtain better recovery performance by using frequency block prior information. The SDP formulation adopted inside each iteration of our new algorithms follows that detailed in [137]. We summarize that SDP formulation in the following paragraph.

Suppose the frequency f of the signal x lies within the frequency block $\mathcal{B} \subset [0, 1]$. Then, given this block prior information $\mathcal{B}$, the atomic norm with block priors and its dual are stated respectively as follows [137, Eq. (II.9)]:

$$\|x\|_{\mathcal{A},\mathcal{B}} = \inf\{\sum_j |c_j| : x = \sum_j c_j a(f_j),\ f_j \in \mathcal{B}\}, \tag{3.76}$$

$$\|q\|^*_{\mathcal{A},\mathcal{B}} = \sup_{\|x\|_{\mathcal{A},\mathcal{B}} \le 1} \langle q, x\rangle_{\mathbb{R}} = \sup_{f \in \mathcal{B}} |\langle q, a(f)\rangle|. \tag{3.77}$$

We formulate the atomic norm minimization with block priors [137, Eq. (III.1)] as

$$\begin{aligned} &\underset{x}{\text{minimize}} \ \ \|x\|_{\mathcal{A},\mathcal{B}} \\ &\text{subject to} \ \ x_j = x_j^\star, \ \ j \in \mathcal{M}. \end{aligned} \tag{3.78}$$

The dual problem of (3.78) is

$$\begin{aligned} &\underset{q}{\text{maximize}}\ \langle q_{\mathcal{M}}, x_{\mathcal{M}} \rangle_{\mathbb{R}} \\ &\text{subject to}\ \ q_{\mathcal{N}\setminus\mathcal{M}} = 0,\ \|q\|^{\star}_{\mathcal{A},\mathcal{B}} \leq 1. \end{aligned} \tag{3.79}$$

Here, $\mathcal{B}$ is a union of disjoint frequency blocks within which all the true frequencies are located, i.e., $f_j^{\star} \in \mathcal{B}$, $\mathcal{B} = \cup_{k=1}^{r}[f_{L_k}, f_{H_k}]$, where r is the number of disjoint block blocks, f_{L_k} and f_{H_k} are the lowest and highest frequencies of the k-th frequency block. Using the properties of positive trigonometric polynomials [74, 81] and (3.77), this dual problem can be formulated as an SDP [137, Eq. (III.16)]:

$$\begin{aligned} &\underset{q,\{\boldsymbol{G}_{a_i}\}_{i=1}^{r},\{\boldsymbol{G}_{b_i}\}_{i=1}^{r}}{\text{maximize}}\ \langle q_{\mathcal{M}}, x_{\mathcal{M}} \rangle_{\mathbb{R}} \\ &\text{subject to}\ \ q_{\mathcal{N}\setminus\mathcal{M}} = 0, \\ &\qquad \delta_{k_i} = \mathcal{L}_{k_i, f_{L_i}, f_{H_i}}(\boldsymbol{G}_{a_i}, \boldsymbol{G}_{b_i}),\ {}^{k_i=0,\ldots,(n-1),}_{i=1,\ldots,r}, \\ &\qquad \begin{bmatrix} \boldsymbol{G}_{a_i} & q \\ q^H & 1 \end{bmatrix} \succeq 0,\ \ i = 1, \ldots, r, \end{aligned} \tag{3.80}$$

where $\delta_{k_i} = 1$ if $k_i = 0$, and $\delta_{k_i} = 0$ otherwise, $\boldsymbol{G}_{a_i} \in \mathbb{C}^{n\times n}$ and $\boldsymbol{G}_{b_i} \in \mathbb{C}^{(n-1)\times(n-1)}$ are Gram matrices. The trace parameterization term $\mathcal{L}_{k,f_L,f_H}(\boldsymbol{G}_a, \boldsymbol{G}_b)$ for the frequency block $[f_L, f_H] \subset [0, 1]$ is set to $\mathrm{tr}[\boldsymbol{\Theta}_k \boldsymbol{G}_a] + \mathrm{tr}[(d_1\boldsymbol{\Theta}_{k-1} + d_0\boldsymbol{\Theta}_k + d_1^H\boldsymbol{\Theta}_{k+1}) \cdot \boldsymbol{G}_b]$, where $\boldsymbol{\Theta}_k$ is the Toeplitz matrix that has ones on the k-th diagonal and zeros elsewhere, $d_0 = -\frac{\alpha\beta+1}{2}$, $d_1 = \frac{1-\alpha\beta}{4} + i\frac{\alpha+\beta}{4}$, where $\alpha = \tan(2\pi f_L/2)$, and $\beta = \tan(2\pi f_H/2)$ when $[f_L, f_H] \subset [0, 0.5]$, and $\alpha = \tan(2\pi(f_L - 1)/2)$, and $\beta = \tan(2\pi(f_H - 1)/2)$ when $[f_L, f_H] \subset (0.5, 1]$. This SDP approach [137] was expanded for more general cases in [138].

Although the atomic norm minimization that exploits external prior information can improve signal recovery performance, in practice, one may not always have direct access to prior information. This leads to the question if we can improve the frequency recovery performance without any external prior information. We describe new algorithms to address this issue in the following section.

3.2.3 Block iterative (re)weighted atomic norm minimization algorithms

We propose three iterative algorithms to enhance frequency recovery performance in the absence of external prior information. In our algorithms, we use estimated frequency support information from previous iterations as block prior for subsequent iterations.

3.2.3.1 Block iterative weighted atomic norm minimization

We first introduce a conceptual algorithm named Block iterative weighted Atomic Norm Minimization (BANM). BANM solves SDP of (3.80) repeatedly, using block priors obtained from the previous iteration. In each iteration, BANM estimates the frequency locations, and then, around the estimated frequencies, BANM forms blocks which very likely contain the true frequencies. With the block priors so obtained, BANM enhances frequency recovery via solving (3.80) in the next iteration, using the new block information.

BANM initially sets the iteration number $t = 0$, frequency block $\mathcal{B} = [0, 1]$, $f \in \mathcal{B}$, and then solves (3.80). Suppose the solution of (3.80) gives r estimated frequencies $f_i^{(t)}$, $i = 1, ..., r$, where the superscript (t) is used to represent the iteration

number. BANM chooses the l frequencies f_{i_1}, f_{i_2}, ..., f_{i_l} with the largest coefficients in amplitude among them, where l is a certain integer number. BANM then forms a union frequency block $\mathcal{B}$ with l frequency subbands around the estimated frequencies $f_{i_j}^{(t)}$, $j = 1, ..., l$, as

$$\mathcal{B} = \bigcup_{j=1}^{l} [f_{i_j}^{(t)} - \tau, f_{i_j}^{(t)} + \tau], \tag{3.81}$$

for some small real number $\tau > 0$, $\tau \in [0, 1]$ that determines the size of the subband. BANM uses the union frequency block $\mathcal{B}$ as block prior and solves (3.80) again with updated parameters. The algorithm continues solving (3.80) and updating (3.81) in each iteration until either a maximum number of iterations is reached or the solution of (3.80) converges.

3.2.3.2 Block iterative reweighted ℓ_1 and atomic norm minimization mixture

BANM requires solving SDP in each iteration to estimate the location of frequencies, but solving SDP repeatedly causes long execution time. Thus, we propose using low-complexity algorithm to obtain prior information on frequency locations. We then use the aforementioned SDP (3.80) only in the last iteration for accurately determining the frequency locations. This concept is the key to design of our algorithm - Block iterative reweighted ℓ_1 and Atomic Norm Minimization Mixture (or simply, BANM-Mix) algorithm that can achieve super-resolution of frequencies with low complexity (Algorithm 3.3).

BANM-Mix first discretizes the continuous frequency domain [0,1] in uniform intervals of size $\triangle_f$. We denote the index set for these intervals as $\mathcal{P} = \{i\}_{i=1}^{p}$, where

$p = 1/\triangle_f$. The index set corresponds to p discrete frequency grid points $f_j = (j-1)/p$, $1 \le j \le p$. We have the discrete Fourier matrix $F \in \mathbb{C}^{n\times p}$ over p discrete frequency grid points whose element in the j-th column and l-th row is $a(f_j)_l = e^{i(2\pi f_j l)}$.

Then, BANM-Mix iteratively solves reweighted ℓ_1 minimization over this discretized frequency dictionary to efficiently estimate frequency locations. Different from iterative reweighted ℓ_1 minimization algorithms designed for incoherent discrete dictionaries [32, 40, 78, 142, 184], our iterative reweighted ℓ_1 minimization algorithm employs novel *adaptive gridding* and *block reweighting* strategies to extract frequency support information from our highly correlated discretized dictionary.

BANM-Mix initializes coefficients $c_i^{(0)} = 0$, weights $w_i^{(0)} = 1$ for $i = 1, ..., p$, and an index set $\mathcal{K}^{(0)} = \{j : j = ql + 1,\ l = 0, 1, ..., (p-1)/q,\ q \in \mathbb{Z}^+\} \triangleq \{j\}_{j=1:q:p} \subseteq \mathcal{P}$. Let $W^{(t)} = diag(w^{(t)}_{\mathcal{K}^{(t)}})$ be a diagonal matrix with weights $w^{(t)}_{\mathcal{K}^{(t)}}$, $F_{\mathcal{K}^{(t)}} \in \mathbb{C}^{|\mathcal{M}|\times|\mathcal{K}^{(t)}|}$ be the partial discrete Fourier matrix.

In the t-th iteration, BANM-Mix solves the following weighted ℓ_1 minimization problem over the index set $\mathcal{K}^{(t)}$, rather than the larger index set $\mathcal{P}$:

$$\begin{aligned} &\underset{z}{\text{minimize}}\ \ \|W^{(t)}z\|_1 \\ &\text{subject to}\ \ x_l^\star = (F_{\mathcal{K}^{(t)}}z)_l,\ \ l \in \mathcal{M}. \end{aligned} \tag{3.82}$$

We then define a vector $c^{(t)}$ having $c^{(t)}_{\mathcal{K}^{(t)}} = z$ and $c^{(t)}_{\mathcal{P}\setminus\mathcal{K}^{(t)}} = 0$.

BANM-Mix then calculates the weight $w_i^{(t+1)}$, $i = 1, ..., p$. We define the index-wise frequency block $\mathcal{BI}_i$ as

$$\mathcal{BI}_i = \{j\ :\ i - b/2 \le j \le i + b/2,\ j \in \mathcal{P}\}, \tag{3.83}$$

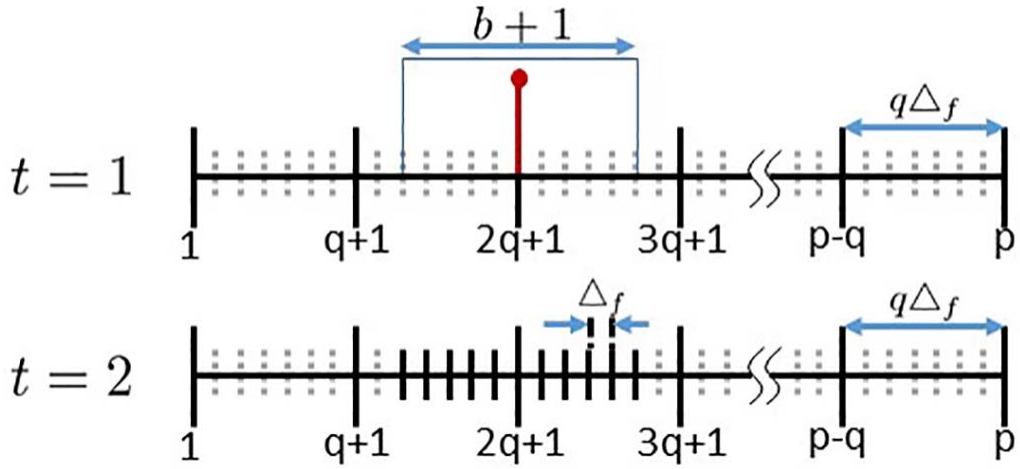


Figure 3.15: An illustration of the adaptive gridding. The estimated frequency f_{2q+1} in the first iteration is depicted by a red pole. The index set $\mathcal{K}$ and $\mathcal{P}$ have solid and dotted grid points respectively. The block that contains the red pole in the middle is the index-wise frequency block $\mathcal{BI}_{2q+1}$, where $b = 10$.

for some positive integer b, which determines the block width. BANM-Mix computes the weight $w_i^{(t+1)}$ by considering the frequency coefficients around f_i in the discretized domain as

$$w_i^{(t+1)} = \frac{1}{\sum_{j \in \mathcal{BI}_i} |c_j^{(t)}| + \epsilon}, \quad i = 1, ..., p, \tag{3.84}$$

where ϵ is a small positive constant to prevent $w_i^{(t+1)}$ from going to infinity. We refer to our procedure in (3.84) as *block reweighting*. Since the discretized dictionary under consideration has highly correlated columns, block reweighting can accurately reflect the likelihood of a true frequency existing around f_i. Our numerical experiments showed that earlier reweighting strategies [32, 40, 142, 184], which update $w_i \leftarrow \frac{1}{|c_i|+\epsilon}$, could not correctly reflect the likelihood of a true frequency being at index i and resulted in worse frequency recovery performance. This is because the solution to (3.82) will disperse the amplitude of a true frequency into the neighboring indices in

highly correlated dictionary columns.

After updating $w_i^{(t+1)}$ for $i = 1, ..., p$, BANM-Mix updates the index set $\mathcal{K}^{(t+1)}$ through adaptive gridding. In adaptive gridding, BANM-Mix first finds indices i, $1 \le i \le p$, with $w_i^{(t+1)} < {}^{(\min(w^{(t+1)})+\max(w^{(t+1)}))}/_2$, where $\min(w^{(t+1)})$ and $\max(w^{(t+1)})$ are the minimum and maximum values of the elements of $w^{(t+1)}$ respectively. We define ${}^{(\min(w^{(t+1)})+\max(w^{(t+1)}))}/_2$ as $w_{\text{mid}}^{(t+1)}$. Then BANM-Mix updates $\mathcal{K}^{(t+1)}$ as

$$\mathcal{K}^{(t+1)} = \mathcal{K}^{(t)} \bigcup \left(\bigcup_{\{i:\, w_i^{(t+1)} < w_{\text{mid}}^{(t+1)},\, i \in \mathcal{P}\}} \mathcal{BI}_i \right). \tag{3.85}$$

Namely, if $w_i^{(t+1)} < w_{\text{mid}}^{(t+1)}$, $\mathcal{K}^{(t+1)}$ will include finer grid points (with separation $\triangle_f$) around frequency $(i-1)/p$. Recall that, at the beginning, $\mathcal{K}^{(0)}$ has only grid points with separation $q\triangle_f$. The reason is that when $w_i^{(t+1)}$ is small, very likely a true frequency exists around frequency $(i-1)/p$. By applying finer gridding around frequency $(i-1)/p$, one can estimate the frequency location more accurately in the next iteration. We call this method of applying different resolutions in the discretized dictionary as *adaptive gridding* (see Figure 3.15).

The algorithm continues solving (3.82) in each iteration until either a specified maximum number of iterations (MaxItr) is exhausted or the solution of (3.82) converges i.e., $||c^{(t-1)} - c^{(t)}||_2 \le \epsilon_{err}$, for some error tolerance $\epsilon_{err} > 0$. BANM-Mix then chooses the block prior set $\mathcal{B}$ by a union of the frequency blocks around frequency $f_i^{(t)}$ satisfying $w_i^{(t+1)} < w_{\text{mid}}^{(t+1)}$. With this frequency block information, we use SDP (3.80) to super-resolve frequencies in the last iteration.

Algorithm 3.3 Block iterative reweighted ℓ_1 and Atomic Norm Minimization Mixture (BANM-Mix) Algorithm

Input: $F \in \mathbb{C}^{n\times p}$, $x^\star_{\mathcal{M}}$, *MaxItr*, b, ϵ, ϵ_{err}
Output: frequency $\hat{f}$, coefficient $\hat{c}$
Initialize: $t \leftarrow 0$, $c^{(t)} \leftarrow 0$, $w^{(t)} \leftarrow 1$, $\mathcal{K}^{(t)} \leftarrow \{i\}_{i=1:q:p}$, $\mathcal{P} \leftarrow \{i\}_{i=1}^{p}$
for $t = 1$ **to** *MaxItr* **do**
  $c^{(t)}_{\mathcal{K}^{(t)}} \leftarrow$ solution of (3.82), $c^{(t)}_{\mathcal{P}\setminus\mathcal{K}^{(t)}} \leftarrow 0$
  $\mathcal{BI}_i \leftarrow$ frequency block via (3.83) for $i = 1, ..., p$
  $w_i^{(t+1)} \leftarrow$ weight via (3.84) for $i = 1, ..., p$
  if $\|c^{(t-1)} - c^{(t)}\|_2 < \epsilon_{err}$ **then**
    $\mathcal{B} \leftarrow$ frequency block via (3.81), where f_i satisfying $w_i^{(t+1)} < (\min(w^{(t+1)}) + \max(w^{(t+1)}))/2$
    $\hat{f} \leftarrow f$ such that $|Q(f)| = 1$ in $\mathcal{B}$ after solving (3.80)
    $\hat{c} \leftarrow c$ satisfying linear equation (3.71) with given $\hat{f}$ and $x^\star_{\mathcal{M}}$
    break
  end
  $\mathcal{K}^{(t+1)} \leftarrow$ index set via (3.85)
end

3.2.3.3 Block iterative reweighted ℓ_1 minimization

The complexity of BANM-Mix can still be high since we have to solve an SDP in the last iteration. To further reduce its complexity, we propose the Block iterative reweighted ℓ_1 Minimization (BL1M) algorithm which is the same as BANM-Mix except that BL1M does not solve SDP in the last iteration. Instead, BL1M uses postprocessing to estimate the final frequencies from the results of iterative reweighted ℓ_1 minimizations. In the last iteration, BL1M finds the frequency blocks $\mathcal{BI}_i$ that satisfy $w_i^{(t+1)} < w_{\text{mid}}^{(t+1)}$. If two frequency blocks $\mathcal{BI}_i$ overlap, BL1M merges them into one. BL1M assumes that one frequency block contains only one true frequency. Suppose that one frequency block (after possible merging) has r grid frequencies $f_1, ..., f_r$ whose corresponding coefficients are $c_1, ..., c_r$. Then BL1M estimates the frequency $\hat{f}$ in that block as $\hat{f} = \frac{\sum_{i=1}^{r} f_i \times |c_i|}{\sum_{i=1}^{r} |c_i|}$.

3.2.4 Numerical experiments

We compare our algorithms with the standard Atomic Norm Minimization (ANM) [168], and the Reweighted Atomic norm Minimization (RAM) [194]. We use CVX [91] to solve convex programs.[3] In all experiments, the phases and frequencies are sampled uniformly at random in $[0, 2\pi)$ and $[0, 1]$ respectively. The amplitudes $|c_j|$, $j = 1, ..., k$, are drawn randomly from the distribution $\sqrt{0.5 + \chi_1^2}$ where χ_1^2 represents the chi-squared distribution with 1 degree of freedom.

We evaluate the recovery performance for the signal dimension $n = 64$, number of observation m is varied from 8 to 25, block width $b = 20$, $\epsilon = 2^8$, and $\epsilon_{err} = 0.5 \times 10^{-4}$. The maximum number of iterations (MaxItr) is set to 20 for both BANM-Mix and RAM.[4] Figure 3.16 and 3.17 show the probability of successful recovery of the entire spectral content over 50 trials for each parameter setup. We consider a recovery successful if $\|f^\star - \hat{f}\|_2 \leq 10^{-3}$. Figure 3.17 clearly shows that our algorithm outperforms both ANM and RAM for $n = 64$ and $k = 8$.

We assess the computational complexity of algorithms in terms of the average execution time for signal recovery from 10 trials. Here, we present results when n is from 120 to 470, $m = \lfloor n/2 \rfloor$, $q = 2^4$, $p = 2^{14}$, $b = 20$, $\epsilon = 2^8$, $\epsilon_{err} = 0.5 \times 10^{-4}$. Figure 3.18 shows that the speed of BL1M is faster than that of ANM and RAM. This is because the latter is based on an SDP while the former uses only ℓ_1 minimization.

---

[3] We conducted our numerical experiments on HP Z220 CMT with Intel Core i7-3770 dual core CPU @3.4GHz clock speed and 16GB DDR3 RAM, using Matlab (R2013b) on Windows 7 OS.

[4] A MaxItr value of 20 was sufficient to guarantee an empirical convergence of our iterative procedures in most of our experiments.

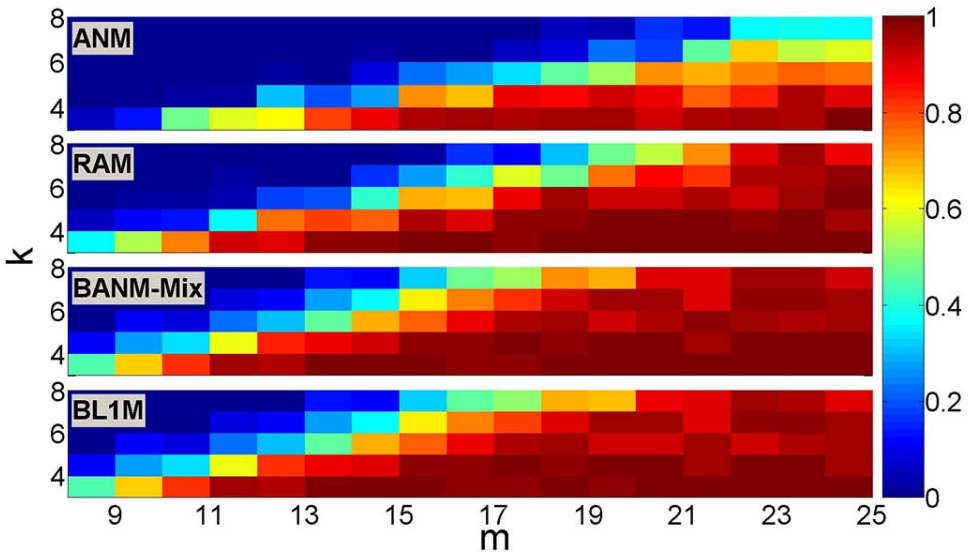


Figure 3.16: The probability P of successful frequency recovery ($n = 64$).

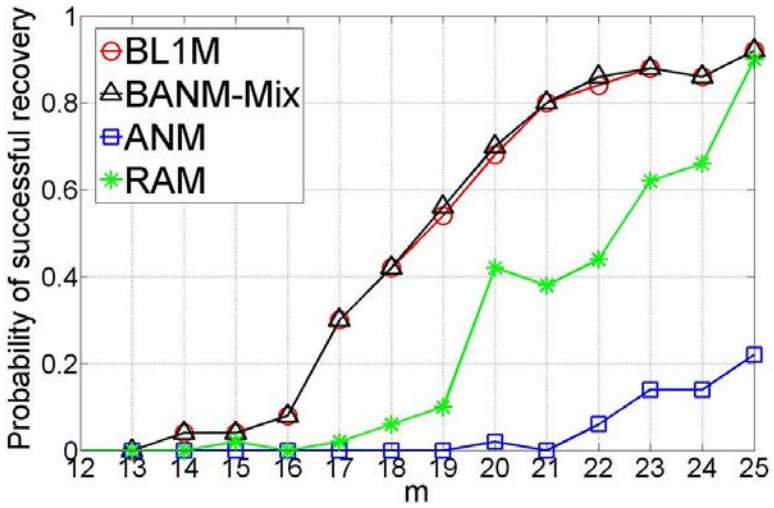


Figure 3.17: The probability P of frequency recovery for $(n, k) = (64, 8)$.

3.2.5 Conclusion

The BANM-Mix and BL1M show better recovery than other known iterative methods [168,194]. In particular, BL1M has shorter execution times than these other methods. Our simulations empirically exhibit convergence of our iterative proce-

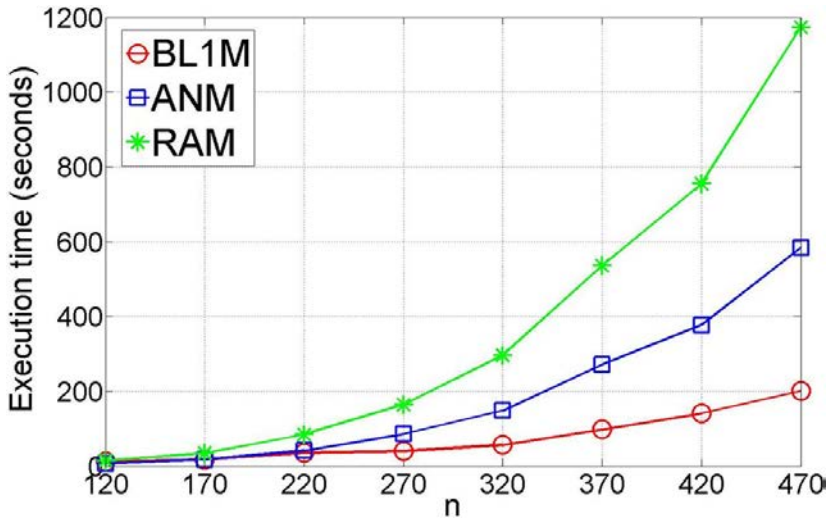


Figure 3.18: The execution time as a function of signal dimension n.

dures. It would be interesting to perform more comprehensive theoretical analysis of convergence in the future.

3.3 Phaseless Super-resolution in the Continuous Domain

3.3.1 Introduction

In engineering and science, improving the accuracy and precision of measurement tools, such as microscopy, X-ray crystallography and MRI, is of great interest. However, due to the physical limitations in measurement tools, sometimes we can only indirectly or partially observe a signal of interest, e.g., obtaining only low-frequency information, only low-resolution image, or only the magnitude of a signal. The microscope is a good example of a measurement tool having such physical limitations ranging from low-frequency measurements to phaseless measurements [11,21,100,165,199].

To overcome the limitation of low-frequency measurements, researchers have

investigated recovering a signal from only its low-frequency Fourier measurements, and referred to it as *super-resolution*. The authors in [27] and [168] proposed SDP based methods for the recovery of signals in the continuous domain under certain separation conditions, by employing *Total Variation Norm Minimization (TVNM)* and *Atomic Norm Minimization (ANM)* respectively. The precise SDP formulation of ANM for d-dimensional signals, $d \geq 2$, was introduced in [188]. Besides, to address the issue of phaseless measurements, people studied *phase retrieval* to obtain phase information from the magnitude measurements of a signal [82,88]. The authors in [25] proposed a trace-norm minimization to solve phase retrieval problem with the use of masks.

Super-resolving a signal from only magnitudes of low-frequency Fourier measurements is often ill-posed due to lack of both phase information and high-frequency information; and hence it is a challenging problem. The authors in [47,105] considered the *phaseless super-resolution* aiming at recovering signals with only low-frequency magnitude measurements. In the noiseless setting, the authors in [47] proposed a combinatorial algorithm for signal recovery using only low-frequency Fourier magnitude measurements, but this algorithm requires additional distinguishing conditions on the signal impulses. In the noisy setting, this combinatorial algorithm suffers from error propagation. Instead of assuming the distinguishing conditions on signals, the authors in [105] used masks to obtain different types of magnitude measurements. The authors provably showed that under appropriate choice of masks, an SDP formulation can be used to recover time-domain impulse signals on the discretized grid.

In this paper, we consider super-resolving time-domain impulse signals located off the grid from only low-frequency Fourier magnitude measurements. To tackle the continuous domain, we propose a novel SDP formulation, employing ANM to recover signals from Fourier magnitude measurements. For example, our approach applies to the magnitude measurements used in [25, 105]. In numerical experiments, we show the successful signal recovery in the continuous domain from only low-frequency magnitude measurements. Also, we compare our method to a simple combining algorithm performing phase retrieval followed by ANM. Our method shows better recovery performance than the simple algorithm.

Notations: In this paper, we denote the set of complex numbers as $\mathbb{C}$. We reserve calligraphic uppercase letters for index sets, e.g., $\mathcal{N}$. We use $|\mathcal{N}|$ as the cardinality of the index set $\mathcal{N}$. We use the superscripts $*$, T, and H to denote conjugate, transpose, and conjugate transpose respectively. We reserve i for the imaginary number, i.e., $i^2 = -1$. We denote a time-domain signal as a lowercase letter, and its frequency-domain signal as its uppercase letter. To denote a ground turth signal, we use the superscript o, e.g., x^o. For the index of a vector and a matrix, we start with the index 0; hence, we denote the first element of the vector X as X_0, and the top-left element of a matrix Q as $Q_{0,0}$.

3.3.2 Problem formulation and background

Let $x^o(t)$ be a sum of Dirac functions expressed as follows:

$$x^o(t) = \sum_{j=1}^{k} c_j^o \delta(t - t_j^o), \tag{3.86}$$

where $\delta(t)$ is the Dirac delta function, $c_j^o \neq 0 \in \mathbb{C}$, and $t_j^o \in [0,1)$. Its Fourier transform is given by:

$$X_f^o = \sum_{j=1}^{k} c_j^o e^{-i2\pi f t_j^o} = \sum_{j=1}^{k} |c_j^o| a(t_j^o, \phi_j^o)_f, \ f \in \mathcal{N}, \tag{3.87}$$

where $f \in \mathcal{N} = \{0, 1, ..., n-1\}$, $a(t_j^o, \phi_j^o) \in \mathbb{C}^{|\mathcal{N}|}$ is an *atom* vector, with the f-th element given by $a(t_j^o, \phi_j^o)_f = e^{-i(2\pi f t_j^o - \phi_j^o)}$. Simply, $X^o = V^o c^o$, where $X^o \in \mathbb{C}^n$, $V^o = [a(t_1^o, 0), ..., a(t_k^o, 0)]$, and $c^o = [|c_1^o| e^{i\phi_1^o}, ..., |c_k^o| e^{i\phi_k^o}]^T$. We also define the minimum separation of $x^o(t)$, denoted by $\triangle_t$, as the closest distance between any two different time value t_j^o's in cyclic manner [27, 168], i.e.,

$$\Delta_t = \min_{t_i^o, t_j^o \in [0,1),\ i \neq j} |t_i^o - t_j^o|. \tag{3.88}$$

The goal here is to find $x^o(t)$ from the low-frequency Fourier magnitude measurements. We state the phaseless super-resolution problem with masks as follows:

$$\begin{aligned} &\text{Find } x(t) \\ &\text{subject to } Z[r,l] = \left| \int_0^1 D_r(t) \sum_{j=1}^{k} c_j^o \delta(t - t_j^o) e^{-i2\pi l t} dt \right|, \\ &\qquad \text{for } -R \leq r \leq R \text{ and } l \in \mathcal{N}, \end{aligned} \tag{3.89}$$

where $Z[r,l]$ is the l-th frequency magnitude obtained by using the r-th mask function $D_r(t)$. Depending on the mask function $D_r(t)$, one can have different types of magnitude information. For example, if we choose $1 + e^{-i2\pi t}$ for $D_r(t)$, we have $|X_l^o + X_{l+1}^o|$, $l \in \mathcal{N}$.

In [105], Jaganathan *et al.* consider the case when the signal $x^o(t)$ is located

on the grid, i.e., $t_j^o \in \{0, 1, 2, ...n-1\}$. By n-point DFT, (3.89) is equivalent to

$$\begin{aligned} &\text{Find } x \\ &\text{subject to } Z[r,l] = |\langle f_l, D_r x\rangle|, \\ &\qquad \text{for } -R \le r \le R \text{ and } l \in \mathcal{N}, \end{aligned} \tag{3.90}$$

where $x \in \mathbb{C}^n$ is a complex valued k-sparse vector, $D_r \in \mathbb{C}^{n\times n}$ is a diagonal matrix, and f_l is the conjugate of the l-th column of the n point DFT matrix. The authors in [105] proposed the following semidefinite relaxation-based program for the phaseless super-resolution in the discrete domain by denoting $Y = xx^H$ and relaxing the rank-1 constraint on Y:

$$\begin{aligned} &\underset{Y}{\text{minimize}} \ \|Y\|_1 + \lambda \text{Tr}(Y) \\ &\text{subject to } Z[r,l]^2 = \text{Tr}(D_r^H f_l f_l^H D_r Y), \\ &\qquad \text{for } -R \le r \le R,\ l \in \mathcal{N}, \text{ and } Y \succeq 0, \end{aligned} \tag{3.91}$$

for some $\lambda > 0$.

This paper makes no assumption of t_j^o being on the grid. In the next section, we propose an ANM based semidefinite relaxation of (3.89) to deal with impulse functions off the grid.

3.3.3 Phaseless super-resolution in the continuous domain

We define the atomic norm of a vector $X \in \mathbb{C}^{|\mathcal{N}|}$ as follows:

$$\|X\|_{\mathcal{A}} = \inf\{\sum_j |c_j| : X_l = \sum_j |c_j| a(t_j, \phi_j)_l, \ {}^{t_j\in[0,1),}_{\phi_j\in[0,2\pi)}\}. \tag{3.92}$$

We have the following new proposition for the atomic norm:

Proposition 3.8. *For any* $X \in \mathbb{C}^{|\mathcal{N}|}$, $\mathcal{N} = \{0, 1, ..., n-1\}$,

$$\|X\|_{\mathcal{A}}^2 = \inf_{u,s} \left\{ \frac{1}{|\mathcal{N}|} s\, Tr(Toep(u)) \; : \; \begin{bmatrix} Toep(u) & X \\ X^H & s \end{bmatrix} \succeq 0 \right\}, \tag{3.93}$$

where $Tr(\cdot)$ *is the trace operator, and* $Toep(u)$ *is the Toeplitz matrix whose first column is* $u = [u_0, u_1, ..., u_{n-1}]^T$. *Moreover, suppose after the Vandermonde decomposition [33, 34, 172],* $Toep(u) = VDV^H$, *where* $V = [a(t_1, 0), ..., a(t_r, 0)]$ *and* D *is a positive diagonal matrix. Then, there exists a vector* c *such that* $X = Vc$ *and* $\sum_j |c_j| = \|X\|_{\mathcal{A}}$.

Proposition 3.8 is similar to Proposition II.1 in [168]; however, Proposition 3.8 considers the trace of $s\text{Toep}(u)$ instead of the sum of trace of $\text{Toep}(u)$ and s. Proposition 3.8 is essential to derive our new SDP formulation handling phaseless measurements.

3.3.3.1 Proof of proposition 3.8

Proof. Let us denote the optimal value of the right hand side of (3.93) by $\text{SDP}(X)$. In order to show $\|X\|_{\mathcal{A}}^2 = \text{SDP}(X)$, we will show that (1) $\|X\|_{\mathcal{A}}^2 \geq \text{SDP}(X)$ and (2) $\|X\|_{\mathcal{A}}^2 \leq \text{SDP}(X)$.

The proof of (1) is easily shown by considering a feasible solution of $\text{SDP}(X)$. For $X = \sum_j |c_j| a(t_j, \phi_j)$, by choosing a feasible solution, $\text{Toep}(u) = \sum_j |c_j| a(t_j, \phi_j) a(t_j, \phi_j)^H$, and $s = \sum_j |c_j|$, we have

$$\begin{bmatrix} \text{Toep}(u) & X \\ X^H & s \end{bmatrix} = \sum_j |c_j| \begin{bmatrix} a(t_j, \phi_j) \\ 1 \end{bmatrix} \begin{bmatrix} a(t_j, \phi_j) \\ 1 \end{bmatrix}^H \succeq 0.$$

For this feasible solution, $\frac{1}{|\mathcal{N}|} s\text{Tr}(\text{Toep}(u)) = (\sum_j |c_j|)^2$, which is $\|X\|_{\mathcal{A}}^2$. Thus, $\text{SDP}(X) \leq \|X\|_{\mathcal{A}}^2$.

For the proof of (2), we will show that for any u, s, and X, $\frac{1}{|\mathcal{N}|}s\mathrm{Tr}(\mathrm{Toep}(u)) \geq \|X\|_{\mathcal{A}}^2$. Suppose for some u, $s \neq 0$, and X, the matrix U in (3.96) is positive semidefinite. From the positive semidefinite condition, we have $\mathrm{Toep}(u) \succeq 0$ and $s > 0$. From the Vandermonde decomposition [33, 34, 172], for any positive semidefinite $\mathrm{Toep}(u)$, we have $\mathrm{Toep}(u) = VDV^H$, where $V = [a(t_1, 0)\ a(t_2, 0), ...a(t_r, 0)]$, and D is a diagonal matrix having d_j as its j-th diagonal element. Since $VDV^H = \sum_{j=1}^{r} d_j a(t_j, 0)a(t_j, 0)^H$ and $\|a(t_j, 0)\|_2^2 = |\mathcal{N}|$, we have $\frac{1}{|\mathcal{N}|}\mathrm{Tr}(\mathrm{Toep}(u)) = \mathrm{Tr}(D)$. Also, from the Vandermonde decomposition and $U \succeq 0$, X is in the range space of V; in fact, if X is not in the range of V, we can always find a vector z such that $z^H U z < 0$. Therefore, $X = Vw = \sum_{j=1}^{r} w_j a(t_j, 0)$, where $w \in \mathbb{C}^r$. By the Schur complement lemma [20], U in (3.96) is expressed as follows:

$$VDV^H - \frac{1}{s}Vww^HV^H \succeq 0. \tag{3.94}$$

It is noteworthy that we can always find a vector q such that $V^H q = sign(w)$, where $sign(w)^H w = \sum_{j=1}^{r} |w_j|$, by choosing $q = V(V^HV)^{-1}sign(w)$. This is because V^H has full row rank. By choosing q such that $V^H q = sign(w)$, we have

$$\mathrm{Tr}(D) = q^H VDV^H q \geq \frac{1}{s}q^H Vww^H V^H q = \frac{1}{s}(\sum_j |w_j|)^2,$$

where the inequality is from (3.94). Therefore, we have

$$\frac{1}{|\mathcal{N}|}s\mathrm{Tr}(\mathrm{Toep}(u)) = s\mathrm{Tr}(D) \geq (\sum_j |w_j|)^2 = \|X\|_{\mathcal{A}}^2.$$

If $s = 0$, from the sufficient and necessary condition for the positive semidefiniteness of a Hermitian matrix, all of U's principal minors need to be non-negative [135]. Thus, $X_j = 0$, $\forall j \in \mathcal{N}$. In this case, Proposition 3.8 still holds. □

3.3.3.2 Lemma for the positive semidefinite matrix Q

Lemma 3.9. *Let* $Q \in \mathbb{C}^{|\mathcal{N}|\times|\mathcal{N}|}$, *and* $X^o \in \mathbb{C}^{|\mathcal{N}|}$. *Suppose (1)* $Q \succeq 0$, *(2)* $Q_{j,j} = Q^o_{j,j}$, $Q_{j,j+1} = Q^o_{j,j+1}$, *and* $Q_{j+1,j} = Q^o_{j+1,j}$, $j \in \mathcal{N} = \{0, 1, ..., n-1\}$, *where* $Q^o = X^o X^{oH}$, *(3)* $X^o_j \neq 0$, $\forall j \in \mathcal{N}$. *Then,* Q *is uniquely determined as* $Q = X^o X^{oH}$.

Proof. From the fact that a Hermitian matrix is positive semidefinite if and only if all of its principal minors are non-negative [135], all of Q's principal minors are required to be non-negative. Let us prove our lemma by induction. When $|\mathcal{N}| = 3$, the determinant of Q is $-|X^o_1 Q_{0,2} - X^o_0 X^o_1 X^{o*}_2|^2$, where $Q_{0,2}$ is unknown. To be $-|X^o_1 Q_{0,2} - X^o_0 X^o_1 X^{o*}_2|^2 \geq 0$, $X^o_1 Q_{0,2} - X^o_0 X^o_1 X^{o*}_2 = 0$. Since $X^o_1 \neq 0$, $Q_{0,2}$ is determined uniquely as $X^o_0 X^{o*}_2$. When $|\mathcal{N}| = 4$, we can consider the top-left 3×3 submatrix of Q and the bottom-right 3×3 submatrix of Q to determine $Q_{0,2}$ and $Q_{1,3}$ respectively. And then, we can deal with 3×3 principal submatrix of Q having $Q_{0,4}$ to determine $Q_{0,4}$. In the similar way, when $|\mathcal{N}| = n$, we can uniquely determine every unknown variables in Q. We omit the detailed explanation due to the space limitation. □

Motivated by Proposition 3.8, we propose the following squared atomic norm minimization for the phaseless super-resolution in the continuous domain, simply *phaseless ANM*:

$$\begin{aligned} &\underset{X}{\text{minimize}} \ \ \|X\|^2_{\mathcal{A}} \\ &\text{subject to} \ \ a_r(X) = b_r, \ r = 1, 2, ..., q, \end{aligned} \tag{3.95}$$

where q is the total number of magnitude measurements, $a_r(X)$ is the magnitude mapping function, $|\langle a_r, X\rangle|$, $a_r \in \mathbb{C}^{|\mathcal{N}|}$, and b_r's are magnitude measurement results.

From Proposition 3.8, we can change (3.95) to

$$\begin{aligned} &\underset{u,X,s}{\text{minimize}} \ \frac{1}{|\mathcal{N}|} s\text{Tr}(\text{Toep}(u)) \\ &\text{subject to } U \triangleq \begin{bmatrix} \text{Toep}(u) & X \\ X^H & \text{s} \end{bmatrix} \succeq 0, \\ &\qquad a_r(X) = b_r, \ r = 1, 2, ..., q, \end{aligned} \tag{3.96}$$

where $u, X \in \mathbb{C}^{|\mathcal{N}|}$ and $s \in \mathbb{C}$. From the positive semidefiniteness of U, $s \geq 0$, and $\text{Toep}(u) \succeq 0$. Besides, if $X_j \neq 0$, $j \in \mathcal{N}$, then $s \neq 0$ from the non-negativeness of all principal minors of U [135]. However, because of the magnitude constraints, (3.96) is a non-convex program.

By the Schur complement lemma [20], $U \succeq 0$ implies $s\text{Toep}(u) - XX^H \succeq 0$. Since $s\text{Toep}(u) = \text{Toep}(su)$, by defining $Q = XX^H$ and $u' = su$, and getting rid of the rank constraint on Q, we have the following SDP relaxation for the phaseless ANM:

$$\begin{aligned} &\underset{Q \succeq 0, u'}{\text{minimize}} \ \frac{1}{|\mathcal{N}|} \text{Tr}(\text{Toep}(u')) \\ &\text{subject to } \ \text{Toep}(u') - Q \succeq 0, \\ &\qquad A_r(Q) = b_r^2, \ r = 1, 2, ..., q, \end{aligned} \tag{3.97}$$

where $A_r(Q)$ is a mapping function, $\text{Tr}(A_r Q)$. Here, $A_r = a_r a_r^H$.

After solving (3.97), we can find the optimal $\hat{Q}$ and optimal $\text{Toep}(\hat{u})$. Our analysis of (3.97) in the following section shows that under certain conditions, $\hat{Q} = X^o X^{oH}$. We can recover X^o up to global phase by the eigenvalue decomposition of $\hat{Q}$. More importantly, because of the structure of $\text{Toep}(\hat{u}) = V^o D V^{oH}$ for some

diagonal matrix D, we can apply any parameter estimation method such as Prony's method [14,112,163] or a matrix pencil method [99,156] to find the time location t_j^o's.

3.3.4 Performance analysis

We first consider the analysis of (3.97) given a rank-1 matrix Q. And then, we provide the analysis of (3.97). Finally, we look at one scenario having magnitude measurements from a set of masks, in which (3.97) provides the desired signal recovery.

Theorem 3.10. *For a given rank-1 positive semidefinite matrix $Q = XX^H$, $X \in \mathbb{C}^{|\mathcal{N}|}$, the following optimization problem provides the squared atomic norm of X, i.e., $\|X\|_{\mathcal{A}}^2$:*

$$\begin{aligned} &\underset{u}{\text{minimize}} \ \frac{1}{|\mathcal{N}|} Tr(Toep(u)) \\ &\text{subject to} \ \ Toep(u) - Q \succeq 0. \end{aligned} \tag{3.98}$$

Proof. We can prove it by using Proposition 3.8. Defining $u = u's$, where $s > 0$ is a scalar. Then we can re-state the constraint as $Toep(u') - \frac{1}{s}XX^H \succeq 0$. By the Schur complement lemma, we have the optimization problem in Proposition 3.8. Therefore, from Proposition 3.8, the optimal value of (3.98) is the same as $\|X\|_{\mathcal{A}}^2$. □

Corollary 3.11. *If (3.97) gives a rank-1 solution to Q, then (3.97) minimizes the squared atomic norm of X among all vectors X satisfying the given constraints $a_r(X) = b_r$, $r = 1, 2, ..., q$.*

Proof. From Theorem 3.10, (3.97) provides the minimum squared atomic norm of X among all vectors X satisfying the constraints $a_r(X) = b_r$, $r = 1, 2, ..., q$. □

Let us consider the case when we have low-frequency Fourier magnitude measurements from a set of masks. The main difference between [27, 168] and our setting is that we have only magnitude measurements instead of measurements offering both phases and magnitudes.

Theorem 3.12. *Given the magnitude measurements* $|X_j^o|$, $|X_j^o + X_{j+1}^o|$, *and* $|X_j^o - iX_{j+1}^o|$, $j \in \mathcal{N} = \{0, 1, ..., n-1\}$, *(3.97) provides the unique solution* $Q = X^o X^{oH}$, *and* $x^o(t)$ *is uniquely obtained up to global phase if the following conditions hold:* $X_j \neq 0$, $\forall j \in \mathcal{N}$, *and* $\Delta_t \geq 4/|\mathcal{N}|$.

Proof. Given magnitude data, $|X_j^o|^2$, $|X_{j+1}^o|^2$, $|X_j^o + X_{j+1}^o|^2$, and $|X_j^o - iX_{j+1}^o|^2$, we can find $Q_{j,j}$, $Q_{j+1,j+1}$, $Q_{j,j+1}$ and $Q_{j+1,j}$, which are the elements of the diagonal, sub-diagonal, and super-diagonal of the matrix Q, by simply solving linear equations on Q together. From Lemma 3.9 in Appendix 3.3.3.2, we can uniquely recover $Q = X^o X^{oH}$ and X^o up to global phase. According to Proposition 3.8 and Theorem 3.10, (3.97) with X^o is essentially the same as the optimization problem dealing with the standard ANM [168] or TVNM [27]. Therefore, (3.97) provides unique $x^o(t)$ up to global phase if the separation condition holds, i.e., $\Delta_t \geq 4/|\mathcal{N}|$. □

3.3.5 Numerical experiments

We compare our phaseless ANM against the standard ANM [168] using measurements offering both phases and magnitudes, as well as against a simple algorithm which first performs the phase retrieval [25] and then applies the standard ANM [168] to recover the impulse functions from the recovered signal using the phase retrieval.

We use CVX [91] to solve (3.97).

Fig. 3.19 (a) and (b) show the probability of successful recovery from the standard ANM and the phaseless ANM respectively. We conducted 50 trials for each parameter setting and measured the success rate. At each trial, we chose one time impulse t_1^o uniformly at random in $[0, 1)$, and another time impulse t_2^o by adding the separation Δ_t to t_1^o in the cyclic manner. We sampled the real part and imaginary part of time coefficients c_j^o's uniformly at random in (0,1). We consider low frequencies, i.e., $\mathcal{M} = \{0, 1, ..., m-1\}$, where $m < n$, $\mathcal{M} \subseteq \mathcal{N}$. The x-axis represents the separation condition Δ_t varied from $1/n$ to $11/n$, and y-axis is the number of low-frequency Fourier measurements m, varied from 2 to 30. For a set of masks in the phaseless ANM, we use the same masks as those of Theorem 3.12 over the index set $\mathcal{M}$. Hence, the number of magnitude measurements is $3m - 2$. We evaluated the recovery performance for the signal dimension $n = 32$. We calculated the Euclidean distance between the estimated and true time locations. If the distance is less than 10^{-3}, then we consider the estimation successful. Numerical experiments show that our phaseless ANM can find the exact time locations in the continuous domain with the same performance as the standard ANM. For large k, e.g., $k = 10$, our method also provides the same performance as the standard ANM. We omit the simulation results in this paper due to the space limitation.

One can think of a simple method conducting the phase retrieval first, and then doing the standard ANM. To compare our algorithm with this simple method, we further carried out numerical experiments by varying the number of magnitude

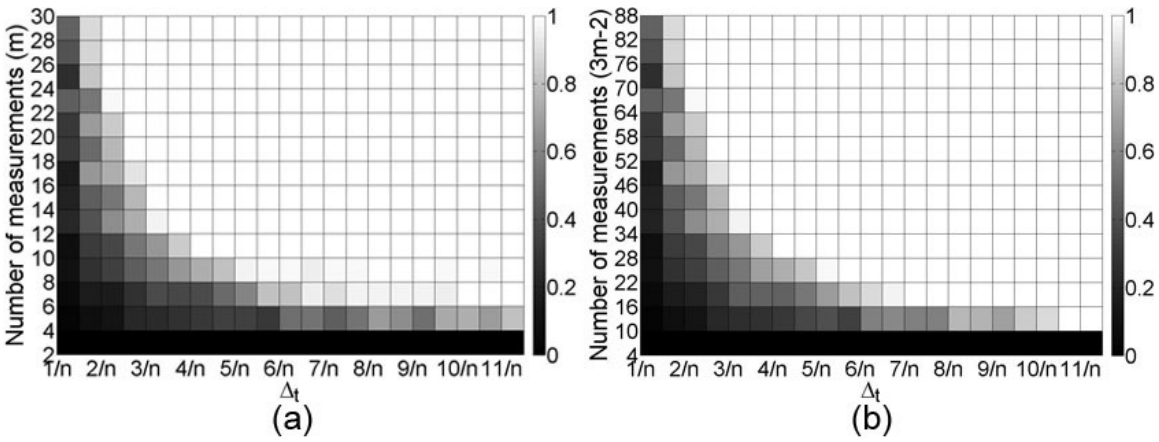


Figure 3.19: The probability P of successful recovery by varying the separation condition Δ_t and the number of measurements when $n = 32$. (a) Standard ANM. (b) Phaseless ANM

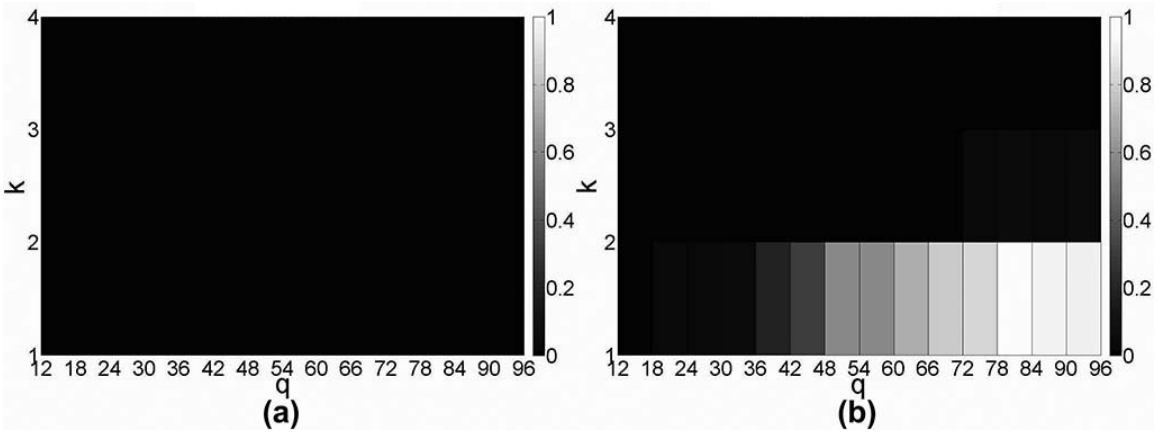


Figure 3.20: The probability P of successful recovery by varying the number of magnitude measurements q and sparsity k when $n = 32$. (a) Phase retrieval and then standard ANM (b) Phaseless ANM

measurements q and the number of sparsity k in (3.86). In this simulation, instead of using a set of masks used in Theorem 3.12, we randomly chose a vector a_r for each magnitude measurement in (3.95). Fig. 3.20 (a) and (b) show the probability of successful recovery from the simple combining algorithm and the phaseless ANM respectively. The x-axis is the number of magnitude measurements q, and y-axis

is the number of sparsity k. With randomly chosen magnitude measurements, our method outperforms the simple combining algorithm.

3.4 Fast Alternating Projected Gradient Descent Algorithms for Recovering Spectrally Sparse Signals

3.4.1 Introduction

Compressed sensing is a signal processing paradigm of reducing the number of measurements needed for accurate signal recovery [31, 66]. Compressed sensing has successfully found applications in various areas such as medical imaging, fluorescence microscopy, face recognition, and radar [77].

Compressed sensing has also found applications in *super-resolution* and *line spectral estimation*, which aims at estimating spectral information from few observations. To recover spectrally sparse signals from uniformly sampled observations, we can apply many conventional techniques, such as Prony's method [150], ESPRIT [155], matrix pencil method [99], and the Tufts and Kumaresan approach [177]. Recently, in [27], Candès and Fernandez-Granda proposed a total variation minimization approach to use a set of *uniformly* spaced time samples to recover sparse continuous-valued frequencies. Tang *et al.* introduced off-the-grid compressed sensing that employs atomic norm minimization to recover sparse continuous-valued frequencies from few *randomly* chosen nonuniformly-spaced time samples in [168]. The total variation minimization or atomic norm minimization employ semidefinite programs (SDP) for recovery. Solving these SDP's is of high complexity, making it difficult to efficiently

recover spectrally sparse signals of high dimensions. Recently, there have been additional efforts to improve the speed or performance of recovering spectrally sparse signals, including grid discretization [167], structured Hankel matrix completion [45], iterative reweighting [51,194], and alternating descent conditional gradient for solving atomic norm minimization [18].

In this paper, we are interested in improving the speed and performance of recovering spectrally sparse signals, especially targeting recovery of high-dimensional spectrally sparse signals. Towards this end, we propose to study a non-convex optimization approach to structured matrix completion. In particular, we consider two non-convex optimization formulations: low-rank *Hankel and Toeplitz Mosaic (HTM)* matrix completion, and low-rank Hankel matrix completion initially introduced in [45]. We suggest general projected Wirtinger gradient [84] descent methods for directly solving these two non-convex optimization formulations, without resorting to convex relaxations. Numerical results show that we can improve the performance or the speed of recovery using projected Wirtinger gradient descent, compared with atomic norm minimization [168] and nuclear norm minimization for Hankel matrix completion [45].

The remainder of this paper is organized as follows. In Section 3.4.2, we briefly review atomic norm minimization [168] and Hankel matrix completion [45]. In Section 3.4.3, we introduce low-rank Hankel and Toeplitz Mosaic (HTM) matrix completion. Thereafter, in Section 3.4.4, we propose projected Wirtinger gradient descent to directly solve the HTM and Hankel completion problems. Finally, in

Section 3.3.5, we demonstrate the efficacy of our algorithms in terms of signal recovery performance as well as the computational complexity.

Notations: We denote the set of complex numbers as $\mathbb{C}$. We reserve calligraphic uppercase letters for index sets. When we use an index set $\mathcal{K}$ as the subscript of a vector x, i.e., $x_{\mathcal{K}}$, it represents the part of the vector x over index set $\mathcal{K}$. We use the superscripts H, T, and $*$ for matrix conjugate transpose, transpose, and conjugate respectively. For $x = [x_1, x_2, ..., x_{2n-1}]^T$, the Hankel matrix over x, denoted $\text{Hank}(x)$, is the Hankel matrix with first column equal to $[x_1, ..., x_n]^T$ and last row equal to $[x_n, ..., x_{2n-1}]$. The Hermitian Toeplitz matrix over x, denoted $\text{Toep}(x)$, has its first column equal to x. We write the Frobenius norm of a matrix $A \in \mathbb{C}^{m\times n}$ as $\|A\|_F \triangleq \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|A_{ij}|^2}$, where A_{ij} is the element of A in its i-th row and j-th column.

3.4.2 Atomic norm minimization and matrix completion

Let $x^\clubsuit$ be a spectrally sparse signal expressed as a sum of k complex exponentials as follows:

$$x_l^\clubsuit = \sum_{j=1}^{k} c_j^\clubsuit e^{i2\pi f_j^\clubsuit l} = \sum_{j=1}^{k} |c_j^\clubsuit| a(f_j^\clubsuit, \phi_j^\clubsuit)_l, \quad l \in \mathcal{N}, \tag{3.99}$$

where $f_j^\clubsuit \in [0,1]$ represents an unknown frequency, $c_j^\clubsuit = |c_j^\clubsuit| e^{i\phi_j^\clubsuit}$ is its coefficient, $\phi_j^\clubsuit \in [0, 2\pi]$ is its phase, and $\mathcal{N} = \{l \ : \ 1 \le l \le 2n-1\}$ is the set of time indices. Here, $a(f_j^\clubsuit, \phi_j^\clubsuit) \in \mathbb{C}^{|\mathcal{N}|}$ is a *frequency-atom*, with l-th element $a(f_j^\clubsuit, \phi_j^\clubsuit)_l = e^{i(2\pi f_j^\clubsuit l + \phi_j^\clubsuit)}$. When the phase is 0, we denote the frequency-atom simply by $a(f_j)$. We assume that the signal is observed over the time index set $\mathcal{M} \subseteq \mathcal{N}$, $|\mathcal{M}| = m \le 2n-1$, where

m observations are chosen randomly. Our goal is to recover the true signal with the smallest possible number of observations. Recovering the true signal is not trivial because the frequencies are in a continuous domain, and their phases and magnitudes are also unknown.

In [168], the authors propose the following atomic norm minimization to recover a spectrally sparse signal $x^{\clubsuit}$ using randomly chosen time samples $\mathcal{M} \subseteq \mathcal{N}$:

$$\begin{aligned} &\underset{x}{\text{minimize}} \ \|x\|_{\mathcal{A}} \\ &\text{subject to} \ \ x_{\mathcal{M}} = x^{\clubsuit}_{\mathcal{M}}, \end{aligned} \tag{3.100}$$

where $\|x\|_{\mathcal{A}}$ represents the atomic norm of a signal x, defined as $\|x\|_{\mathcal{A}} = \inf\{\sum_j |c_j| : x = \sum_j c_j a(f_j)\}$. The atomic norm minimization (3.100) can be cast as an SDP as follows [168, (II.6)]:

$$\begin{aligned} &\underset{u,x,t}{\text{minimize}} \ \ \frac{1}{2|\mathcal{N}|}\text{Tr}(\text{Toep}(u)) + \frac{1}{2}t \\ &\text{subject to} \ \ U_1 = \begin{bmatrix} \text{Toep}(u) & x \\ x^H & t \end{bmatrix} \succeq 0, \\ &\qquad\qquad x_{\mathcal{M}} = x^{\clubsuit}_{\mathcal{M}}, \end{aligned} \tag{3.101}$$

where $\text{Tr}(\cdot)$ is the trace function, and $\text{Toep}(u)$ is a Hermitian Toeplitz matrix. We refer the reader to [168] for details. The atomic norm minimization requires a certain minimum separation between frequencies for successful recovery.

Inspired by the matrix pencil approach [98, 99], the authors of [45] developed the Enhanced matrix completion (EMaC) to recover spectrally sparse signals from

randomly chosen time samples. The EMaC formulation is stated as follows [45, (13)]:

$$\begin{aligned} &\underset{x}{\text{minimize}} \ \ \|\text{Hank}(x)\|_* \\ &\text{subject to} \ \ x_{\mathcal{M}} = x^{\clubsuit}_{\mathcal{M}}, \end{aligned} \tag{3.102}$$

where $\|M\|_*$ represents the nuclear norm of a matrix M, which is the sum of the singular values of M. This convex program can be further as an SDP:

$$\begin{aligned} &\underset{x,Q_1,Q_2}{\text{minimize}} \ \ \frac{1}{2}\text{Tr}(Q_1) + \frac{1}{2}\text{Tr}(Q_2) \\ &\text{subject to} \ \ U_2 = \begin{bmatrix} Q_1 & \text{Hank}(x) \\ \text{Hank}(x)^H & Q_2 \end{bmatrix} \succeq 0, \\ &\qquad\qquad x_{\mathcal{M}} = x^{\clubsuit}_{\mathcal{M}}, \end{aligned} \tag{3.103}$$

where the matrices Q_1 and Q_2 are Hermitian matrices.

When the underlying frequencies satisfy the separation condition, atomic norm minimization has better signal recovery performance than EMaC. However, EMaC provides robust signal recovery even when the separation condition is not satisfied. Moreover, solving the SDP for atomic norm minimization and EMaC can be slow, especially for problems of large dimensions. In the next section, we propose new optimization formulations and algorithms which can provide better recovery performance of spectrally sparse signals in faster speed, compared with atomic norm minimization and nuclear norm minimization for Hankel matrix completion.

3.4.3 Hankel and Toeplitz mosaic matrix completion

In this section, we introduce a new optimization formulation, called Hankel and Toeplitz Mosaic (HTM) matrix completion, to recover spectrally sparse signals. Our HTM matrix completion is formulated as follows:

$$\begin{aligned} &\underset{z,x}{\text{minimize}} \ \ \text{rank}(U_3) \\ &\text{subject to} \ \ U_3 = \begin{bmatrix} \text{Toep}(z) & \text{Hank}(x) \\ \text{Hank}(x)^H & \text{Toep}(z)^T \end{bmatrix} \succeq 0, \\ &\qquad\quad x_{\mathcal{M}} = x^{\clubsuit}_{\mathcal{M}}, \end{aligned} \tag{3.104}$$

where $\text{rank}(\cdot)$ denotes the matrix rank. One can relax (3.104) to its nuclear norm minimization:

$$\begin{aligned} &\underset{z,x}{\text{minimize}} \ \ 2\text{Tr}(\text{Toep}(z)) \\ &\text{subject to} \ \ U_3 = \begin{bmatrix} \text{Toep}(z) & \text{Hank}(x) \\ \text{Hank}(x)^H & \text{Toep}(z)^T \end{bmatrix} \succeq 0, \\ &\qquad\quad x_{\mathcal{M}} = x^{\clubsuit}_{\mathcal{M}}. \end{aligned} \tag{3.105}$$

The difference between (3.103) and (3.105) is that (3.105) further imposes that the diagonal matrices are Hermitian Toeplitz matrices.

The following theorem states a relation between (3.105) and atomic norm minimization.

Theorem 3.13. *Let $x^\dagger$ be an optimal solution to (3.105), and suppose that Hank$(x^\dagger)$ can be decomposed as VDV^T, where D is a diagonal matrix and V has frequency atoms*

as its columns. Then the optimal objective value of (3.105) is the smallest atomic norm among all signals x satisfying the observation constraints; namely (3.105) is equivalent to atomic norm minimization.

Proof. From the positive semidefiniteness of U_3, $\text{Toep}(\mathbf{z}) \succeq 0$. By the Vandermonde decomposition [33, 34, 172], for any positive semidefinite $\text{Toep}(\mathbf{z})$, we have $\text{Toep}(\mathbf{z}) = \tilde{V}\tilde{D}\tilde{V}^H$, where $\tilde{V} = [a(f_1, 0)\ a(f_2, 0), ...a(f_r, 0)]$, and $\tilde{D}$ is a diagonal matrix having $d_j > 0$ as its j-th diagonal element. Since $\tilde{V}\tilde{D}\tilde{V}^H = \sum_{j=1}^{r} d_j a(f_j, 0)a(f_j, 0)^H$ and $\|a(f_j, 0)\|_2^2 = |\mathcal{N}|$, we have $\frac{1}{|\mathcal{N}|}\text{Tr}(\text{Toep}(\mathbf{z})) = \text{Tr}(D)$. Also, from $\text{Hank}(\mathbf{x}) = VDV^T$, if $\tilde{V}$ is in different range space from V, then, we can always find a vector $\mathbf{z}$ such that $\mathbf{z}^H U_3 \mathbf{z} < 0$. Hence, we have $\text{Toep}(\mathbf{z}) = V\tilde{D}V^H$, and the following condition:

$$\begin{bmatrix} \text{Toep}(\mathbf{z}) & \text{Hank}(\mathbf{x}) \\ \text{Hank}(\mathbf{x})^H & \text{Toep}(\mathbf{z})^T \end{bmatrix} = \begin{bmatrix} V\tilde{D}V^H & VDV^T \\ V^*DV^H & V^*\tilde{D}V^T \end{bmatrix} = \begin{bmatrix} V & 0 \\ 0 & V^* \end{bmatrix} \begin{bmatrix} \tilde{D} & D \\ D & \tilde{D} \end{bmatrix} \begin{bmatrix} V & 0 \\ 0 & V^* \end{bmatrix}^H \succeq 0.$$

Since (3.105) provides the minimum sum of $d_j > 0$'s, $\text{Toep}(\mathbf{z})$ becomes the same as $\|x\|_{\mathcal{A}}$. □

3.4.4 Projected Wirtinger gradient descent for HTM and Hankel matrix completion

We now propose a projected Wirtinger gradient descent algorithm for various structured matrices, especially, HTM matrix completion and Hankel matrix completion. We first introduce the method for HTM matrix completion, which solves (3.104).

Let us define the set of all complex-valued matrices with rank no greater than

k as follows:

$$\mathcal{R}_{\mathbb{C}}^{k} = \{L \in \mathbb{C}^{2n\times 2n} \mid \operatorname{rank}(L) \leq k\}. \tag{3.106}$$

We further denote the set of all complex-valued Toeplitz Hankel mosaic matrices that are consistent with the observed data as

$$\mathcal{H}_{HTM} = \left\{ \begin{bmatrix} \operatorname{Toep}(z) & \operatorname{Hank}(x) \\ \operatorname{Hank}(x)^H & \operatorname{Toep}(z)^T \end{bmatrix} \;\middle|\; z \in \mathbb{C}^n,\ x \in \mathbb{C}^{2n-1},\ x_{\mathcal{M}} = x_{\mathcal{M}}^{\clubsuit} \right\}. \tag{3.107}$$

We then seek a matrix in $\mathcal{H}_{HTM}$ with rank no larger than k by considering the problem:

$$\underset{L\in\mathcal{R}_{\mathbb{C}}^{k}, H\in\mathcal{H}}{\text{minimize}} \ \frac{1}{2}\|L-H\|_F^2. \tag{3.108}$$

To solve (3.108), we use a projected gradient descent algorithm. We start with initial values H_0 and L_0. By considering Wirtinger calculus [84] and applying gradient descent on the function with complex variables, we derive the update rule of our projected Wirtinger algorithm as follows:

$$\begin{aligned} L_{t+1} &\in \mathcal{P}_{\mathcal{R}_{\mathbb{C}}^{k}}(L_t - \delta_1(L_t - H_t)), \\ H_{t+1} &\in \mathcal{P}_{\mathcal{H}_{HTM}}(H_t - \delta_2(H_t - L_{t+1})), \end{aligned} \tag{3.109}$$

where t is the iteration number, $\delta_1 > 0$ and $\delta_2 > 0$ are step sizes, and $\mathcal{P}_{\mathcal{R}_{\mathbb{C}}^{k}}$ and $\mathcal{P}_{\mathcal{H}_{HTM}}$ are the projections onto $\mathcal{R}_{\mathbb{C}}^{k}$ and $\mathcal{H}_{HTM}$ respectively.

The projection $\mathcal{P}_{\mathcal{R}_{\mathbb{C}}^{k}}(X)$, which is the projection onto the best rank-k approximation to X, is calculated through singular value decomposition. Let the singular value decomposition of X be $X = U\Lambda V^H$. Then, $\mathcal{P}_{\mathcal{R}_{\mathbb{C}}^{k}}(X) = U_k\Lambda_k V_k^H$, where Λ_k is

Algorithm 3.4 Projected Wirtinger gradient descent method

Input: $x^{\clubsuit}_{\mathcal{M}}$, $MaxItr$, ϵ_{err}, δ_1, δ_2, k
Output: x
Initialize: $L_0 \leftarrow 0$, $H_0 \leftarrow 0$
for $t = 0$ **to** $MaxItr$ **do**
  $L_{t+1} \leftarrow \mathcal{P}_{\mathcal{R}^k_{\mathbb{C}}}(L_t - \delta_1(L_t - H_t))$
  $H_{t+1} \leftarrow \mathcal{P}_{\mathcal{H}}(H_t - \delta_2(H_t - L_{t+1}))$ ▷ $\mathcal{H}$ is $\mathcal{H}_{HTM}$ for HTM completion, or $\mathcal{H}$ is the set of Hankel matrices for Hankel matrix completion
  if $\|H_{t+1} - H_t\|_F / \|H_t\|_F < \epsilon_{err}$ **then**
    break
  end
end

the diagonal matrix that only retains the k largest nonnegative singular values of X, and U_k and V_k are the matrices whose columns are the corresponding left and right singular vectors respectively.

The projection $\mathcal{P}_{\mathcal{H}_{HTM}}(X)$ is carried out for Hankel and Toeplitz matrices separately due to its mosaic structure. More precisely, for any $X = \begin{bmatrix} X_{11} & X_{12} \\ X_{12}^* & X_{22} \end{bmatrix} \in \mathbb{C}^{2n\times 2n}$, we have

$$\mathcal{P}_{\mathcal{H}_{HTM}}(X) = \begin{bmatrix} \frac{1}{2}\mathcal{P}_T(X_{11} + X_{22}^T) & \mathcal{P}_H(X_{12}) \\ \mathcal{P}_H(X_{12})^H & \frac{1}{2}\left(\mathcal{P}_T(X_{11} + X_{22}^T)\right)^T \end{bmatrix}.$$

Here $\mathcal{P}_H$ is the projection onto the set of Hankel matrices whose anti-diagonals coincide with the observed data. From [23, (8)],

$$\mathcal{P}_H(Y) = \text{Hank}(z),$$
$$\text{where } \begin{cases} z_i = y_i, & i \in \mathcal{M}, \\ z_i = \text{mean}\{Y_{ab} | a + b = i\}, & \text{otherwise.} \end{cases} \tag{3.110}$$

Namely, for the missed measurement signal z_i, $i \in \mathcal{N} \setminus \mathcal{M}$, we calculate the average of the i-th anti-diagonal elements of Y. The operator $\mathcal{P}_T$ is the projection onto the set of Toeplitz matrices, and is given by

$$\begin{aligned} &\mathcal{P}_T(Y) = \text{Toep}(z), \\ &\text{where } z_i = \text{mean}\{Y_{ab} | a - b = i, a \geq b\}. \end{aligned} \tag{3.111}$$

Namely, for Toeplitz matrix projection, we compute the average of the i-th diagonal elements. The projected Wirtinger algorithm continues the projections as in (3.109) onto a low-rank matrix and Hankel Toeplitz mosaic matrix until it converges to a solution or the maximum number of iterations (MaxItr) is exhausted.

The projected Wirtinger for Hankel matrix completion is similar to the projected Wirtinger for HTM. It uses the projection onto the Hankel matrix, instead of Hankel Toeplitz mosaic matrix, according to (3.110). Based on Attouch and Bolte's theory [5, 16], we provide global convergence analysis of the algorithm in [23]. We refer the reader to our longer version [23] for details. We summarize our procedure in Algorithm 3.4.

3.4.5 Numerical experiments

We compare the performance and complexity of our projected Wirtinger gradient descent methods for both HTM and Hankel matrix completion with the standard Atomic Norm Minimization (ANM) [168], and the Enhanced Matrix Completion (EMaC) [45]. We conducted our numerical experiments on HP Z220 CMT with Intel Core i7-3770 dual core CPU @3.4GHz clock speed and 16GB DDR3 RAM, using Mat-

lab (R2013b release) on Windows 7 OS. We use CVX [91] to solve convex programs for ANM and EMaC. In all experiments, the phases and frequencies are sampled uniformly at random in $[0, 2\pi)$ and $[0, 1]$ respectively. The amplitudes $|c_j|$, $j = 1, ..., k$, are chosen as $|c_j| = 1+10^{0.5m_j}$, where m_j is randomly drawn in the uniform distribution on [0,1].

We carried out numerical experiments to compare the signal recovery performance of our projected Wirtinger method with ANM and EMaC. We abbreviate HTM and Hankel matrix completion using projected Wirtinger to HTM-PW and EMaC-PW respectively. We varied the number of measurements m from 20 to 127, and the sparsity k from 1 to 40. We obtained the probability of successful signal recovery over 100 trials for each parameter setup. We consider a recovery successful if $\|x^\clubsuit - \hat{x}\|_2/\|x^\clubsuit\|_2 \leq \epsilon_{succ}$, where $\epsilon_{succ} = 10^{-2}$, $\hat{x}$ is the estimated signal, and $x^\clubsuit$ is the true signal. Fig. 3.21 demonstrates that the projected Wirtinger algorithms (Fig. 3.21(a) and (b)) improve the phase transition boundary over other algorithms (Fig. 3.21(c) and (d)). We provide Fig. 3.22 to more clearly show the advantage of our algorithms in signal recovery, where we use the success criterion $\epsilon_{succ} = 5 \times 10^{-3}$. In particular, when the sparsity k is 40, HTM-PW has 50% success rate with around 87 measurements out of 127 respectively, while the success rate of other algorithms with the same number of measurements is 0. The 50% success rate of other algorithms is achieved around 110 for EMaC, and 120 for ANM. Even though HTM-PW has the best phase transition boundary curve, in certain regions (upper orange color region) of its phase transition, HTM-PW has smaller success rate than other algorithms. It

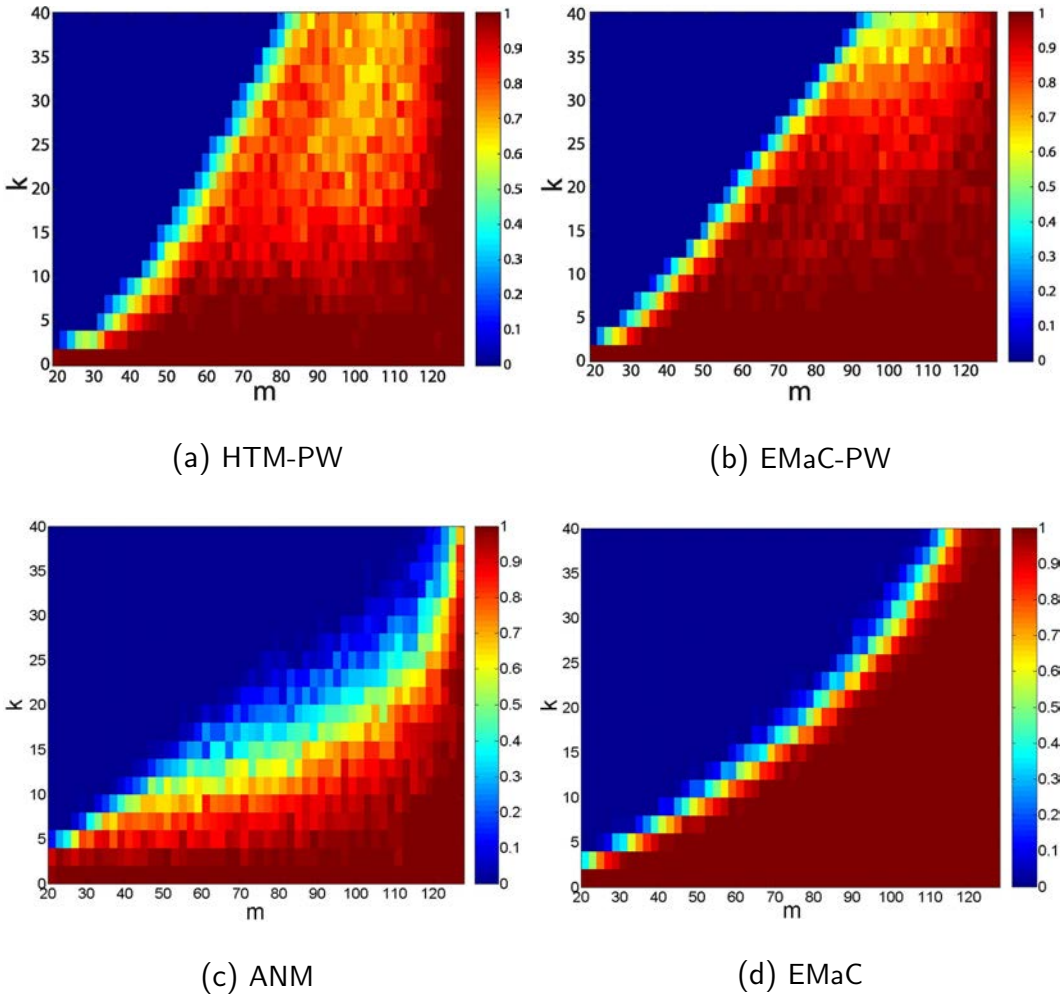


Figure 3.21: Phase transition for successful signal recovery when the signal dimension $2n - 1 = 127$.

would be interesting to understand this phenomenon more deeply. We leave this for future work.

We assess the computational complexity of our algorithms in terms of the average execution time by averaging over 10 trials. We provide results in Table 3.1 when the signal dimension $(2n - 1)$ varies from 101 to 1401, $m = \lfloor (2n-1)/2 \rfloor$, and $k = \lfloor 0.1(2n - 1) \rfloor$. We stopped our projected Wirtinger algorithms when $\|H_{t+1} -$

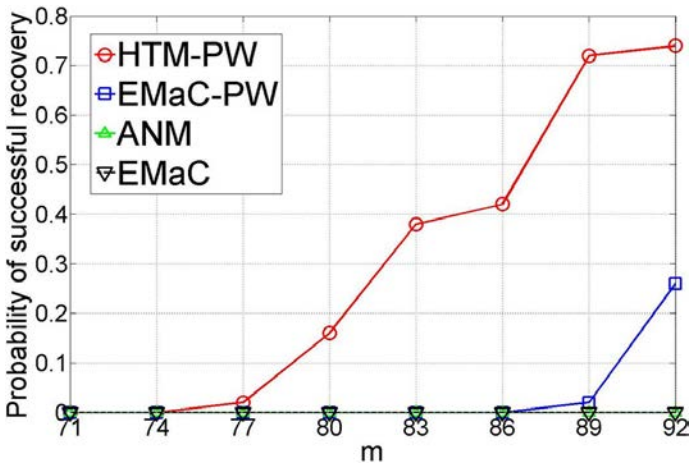


Figure 3.22: The probability of signal recovery when $(2n-1, k) = (127, 40)$.

Table 3.1: Execution Time (Unit: seconds)

$2n-1$	Project Wirtinger		CVX	
	HTM	EMaC	ANM	EMaC
101	3.7	0.1	27.1	5.7
201	7.1	0.2	787.7	51.6
401	309.7	0.9	N/A[a]	N/A
601	733.4	1.4	N/A	N/A
1001	3612.4	6.6	N/A	N/A
1401	8999.2	10.1	N/A	N/A

[a] Not Available (Out of Memory)

$H_t||_F/||H_t||_F \leq 10^{-6}$ or the maximum number of iterations (MaxItr) is exhausted. We set MaxItr to 1000. The success rate was similar to that shown in Fig. 3.21. Table 3.1 clearly shows that the speed of our projected Wirtinger methods outperform those of ANM and EMaC.

CHAPTER 4
COMPRESSED HYPOTHESIS TESTING

4.1 Introduction

In many areas of science and engineering such as network tomography, cognitive radio, and radar, one needs to infer statistical information of signals of interest. Statistical information of interest can be the mean, variance or even distributions of certain random variables [56, 73, 107, 118, 127, 146, 147, 186]. Obtaining such statistical information is essential in detecting anomalous behaviors of random signals. For example, inferring distributions of random variables has important applications in quickest detections of potential hazards, in detecting changes in statistical behaviors of random variables [9, 118, 147], and also in detecting congested links with abnormal delay statistics in network tomography [186].

This paper focuses on the anomaly detection problem. In particular, we consider n random variables, denoted by X_i, $i \in \mathcal{S} = \{1, 2, ..., n\}$, out of which k ($k \ll n$) random variables follow a probability distribution $f_2(\cdot)$ while the much larger set of remaining $(n - k)$ random variables follow another probability distribution $f_1(\cdot)$. However, it is unknown which k random variables follow the distribution $f_2(\cdot)$. Our goal is to infer the subset of random variables that follow $f_2(\cdot)$. In our problem setup, this is equivalent to determining whether X_i follows the probability distribution $f_1(\cdot)$ or $f_2(\cdot)$ for each i. This system model of anomaly detection has appeared in various applications, such as [118, 119, 130–132, 147].

In order to infer the probability distribution of these n random variables, one conventional method is to get l separate samples for each random variable X_i and then use hypothesis testing techniques to determine whether X_i follows the probability distribution $f_1(\cdot)$ or $f_2(\cdot)$ for each i. To ensure correctly identifying the k anomalous random variables with a high probability, at least $\Theta(n)$ samples are needed for hypothesis testing using these samples involving only individual random variables. However, when the number of random variables n grows large, the requirement on sampling rates and sensing resources can be tremendous. For example, in a sensor network, if the fusion center aims to track the anomalies in data generated by n chemical sensors, sending all the data samples of individual sensors to the fusion center will be energy-consuming for the energy-limited sensor network. In such scenarios, we would like to infer the probability distributions of the n random variables with as few data samples as possible. In some applications, due to physical constraints [73, 127, 186], we cannot directly get separate samples of individual random variables. Those difficulties raise the question of whether we can perform hypothesis testing from a much smaller number of samples involving mixed observations.

One way to achieve hypothesis testing with a smaller number of samples is to utilize the sparsity of anomalous random variables, that is the number of anomalous random variables k is much smaller than the total number of random variables n. By utilizing the sparsity of anomalous random variables, [35,130–132] optimized adaptive *separate* samplings of individual random variables and reduced the number of needed samples for individual random variables. It is worth noting that the total number of

observations is still at least $\Theta(n)$ for these methods, if one is restricted to sample the n random variables individually [35, 130–132].

In this paper, we propose a new approach, named *compressed hypothesis testing*, to find the k anomalous random variables from non-adaptive *mixed* observations of n random variables. In this new approach, instead of a separate observation of an individual random variable, we make each observation a function of the n random variables. Our hypothesis testing method is motivated by compressed sensing [31, 67, 68], which is the technique that recovers a *deterministic* sparse vector from its linear projections. Our analysis shows that our new approach can reduce the number of samples required for reliable hypothesis testing. In particular, we show that the number of samples needed to correctly identify the k anomalous random variables can be reduced to $O\left(\frac{k\log(n)}{\min_{p_v,p_w} C(p_v,p_w)}\right)$ observations, where $C(p_v,p_w)$ is the Chernoff information between two possible distributions p_v and p_w for the proposed mixed observations. We also show that mixed observations can strictly increase error exponents of the hypothesis testing, compared with separate sampling of individual random variables. For special cases of Gaussian random variables, we derive optimal mixed measurements to maximize the error exponent of the hypothesis testing. To reduce the computational complexity, we further design efficient algorithms - message passing (MP) based algorithm and Least Absolute Shrinkage and Selection Operator (LASSO) based algorithm - to infer anomalous random variables from mixed observations. We also provide intensive numerical examples to illustrate the advantage of the proposed approach.

This paper is organized as follows. In Section 4.2, we introduce the mathematical models for the considered anomaly detection problem. In Section 4.3.1, we investigate hypothesis testing error performance using time-invariant mixed observations, propose corresponding hypothesis testing algorithms and provide their performance analysis. In Section 4.3.2, we consider using random time-varying mixed observations to identify the anomalous random variables, and derive the error exponent of wrongly identifying the anomalous random variables. In Section 4.3.3, we consider using deterministic time-varying mixed observations for hypothesis testing, and derive a bound on the error probability. In Section 4.4, we demonstrate, by examples of Gaussian random variable vectors, that linear mixed observations can strictly improve the error exponent over separate sampling of each individual random variables. In Section 4.5, we derive the optimal mixed measurements for Gaussian random variables maximizing the hypothesis testing error exponent. In Section 4.6, we introduce efficient algorithms to find abnormal random variables using mixed observations, for large values of n and k. In Section 4.7, we provide numerical results to demonstrate the effectiveness of our hypothesis testing method from linear measurements. Section 4.8 describes the conclusion of this paper.

4.2 Mathematical Models

We consider n independent random variables X_i, $i = 1, \cdots, n$. Out of these n random variables, $(n-k)$ of them follow a known probability distribution $f_1(\cdot)$; while

the other k random variables follow another known probability distribution $f_2(\cdot)$:

$$X_i \sim \begin{cases} f_1(\cdot), & i \notin \mathcal{K} \\ f_2(\cdot), & i \in \mathcal{K}, \end{cases} \tag{4.1}$$

where $\mathcal{K} \subset \mathcal{S}$ is an unknown "support" index set, and $|\mathcal{K}| = k \ll n$.

Our goal is to determine $\mathcal{K}$ by identifying those k anomalous random variables with as few samples as possible. We take m mixed observations of the n random variables at m time indices. The measurement at time j is stated as

$$Y^j = g^j(X_1^j, X_2^j, ..., X_n^j),$$

which is a function of n random variables, where $1 \le j \le m$. Note that the realization x_i^j of the random variable X_i^j and the random variable X_i^j follow the same distribution as the random variable X_i. We assume that the realizations at different time slots are mutually independent. Although our results can be extended to nonlinear observations, in this paper, we specifically consider the case when the functions g^j's are *linear* due to its simplicity and its wide range of applications. When the functions g^j's are linear, the j-th measurement is stated as follows:

$$Y^j = g^j(X_1^j, X_2^j, ..., X_n^j) = \sum_{i=1}^{n} a_i^j X_i^j = \langle a^j, X^j \rangle, \tag{4.2}$$

where a sensing vector $a^j = [a_1^j, a_2^j, \ \dots \ , a_n^j]^T \in \mathbb{R}^{n \times 1}$, and $X^j = [X_1^j, X_2^j, ..., X_n^j]^T$. We obtain an estimate of the index set using a decision function ϕ from Y^j, $j = 1, \cdots, m$, as follows:

$$\hat{\mathcal{K}} = \phi(Y^1, \cdots, Y^m). \tag{4.3}$$

We would like to design the sampling functions g^j's and the decision function ϕ such that the probability

$$P(\hat{\mathcal{K}} \neq \mathcal{K}) \leq \epsilon, \tag{4.4}$$

for an arbitrarily small $\epsilon > 0$.

Our approach is motivated by compressed sensing technique. However, our approach has a major difference from compressed sensing. In our setup, each random variable X_i takes an *independent* realization in each measurement, while in the conventional compressed sensing problem $y = Ax$, where y is the observation vector $y \in \mathbb{R}^{m\times 1}$ and $A \in \mathbb{R}^{m\times n}$ is a sensing matrix, the vector $x = [x_1, x_2, x_3, ..., x_n]^T$ takes the same *deterministic* values across all the measurements. In some sense, our problem is a probabilistic "random-variable" generalization of the compressed sensing problem. In compressed sensing, Bayesian compressed sensing stands out as one model where prior probability distributions of the vector x is considered [108,195]. However, in [108, 195], the vector $x = [x_1, x_2, x_3, ..., x_n]^T$ is *fixed* once the random variables are realized from the prior probability distribution, and then remains *unchanged* across different measurements. That is fundamentally different from our setting where random variables dramatically change across different measurements. There also exist a collection of research works discussing compressed sensing for *smoothly* time-varying signals [4, 111, 180]. In contrast, the objects of interest in this research are random variables taking *completely independent* realizations at different time indices, and we are interested in recovering statistical information of random variables, rather than recovering the deterministic values.

Table 4.1: Three different types of mixed observations

Measurement type	Definition
Fixed time-invariant	The measurement function is the same at every time index.
Random time-varying	A measurement function is randomly generated from a distribution at each time index.
Deterministic time-varying	A measurement function is predetermined, but changed at each time index to measure a fixed fraction of different random variables at different time.

Notations: We denote a random variable and its realization by an uppercase letter and the corresponding lowercase letter respectively. We use X_i to refer to the i-th element of the random variable vector X. We reserve calligraphy uppercase letters $\mathcal{S}$ and $\mathcal{K}$ for index sets, where $\mathcal{S} = \{q : 1 \leq q \leq n\} = \{q\}_{q=1}^{n}$, and $\mathcal{K} \subseteq \mathcal{S}$. We use superscripts to represent time indices. Hence, x^j represents the realization of a random variable vector X at time j. We reserve the lowercase letters f and p for probability density functions. We denote the probability density function $p_X(x)$ as $p(x)$ or p_X. We abuse this notation for simplicity and helping understanding.

4.3 Compressed Hypothesis Testing

In compressed hypothesis testing, we consider three different types of mixed observations, namely *fixed time-invariant* mixed measurements, *random time-varying* measurements, and *deterministic time-varying* measurements. Table 4.1 shows the definition of these types. For these different types of mixed observations, we characterize the number of measurements required to achieve a specified hypothesis testing error probability.

4.3.1 Fixed time-invariant measurements

In this subsection, we focus on a simple case in which sensing vectors are time-invariant across different time indices, i.e., $a^1 = \cdots = a^m \triangleq a$, $a \in \mathbb{R}^{n\times 1}$. This simple case helps us to illustrate the main idea that will be generalized to more sophisticated schemes in later sections.

We first give the likelihood ratio test algorithm in Algorithm 4.1, over $l = \binom{n}{k}$ possible hypotheses. To analyze the number of required samples for achieving a certain hypothesis testing error probability, we consider another related hypothesis testing algorithm based on pairwise hypothesis testing in Algorithm 4.2, which is suboptimal compared to the likelihood ratio test algorithm. There are l possible probability distributions for the output of the function $g(\cdot)$, depending on which k random variables are anomalous. We denote these possible distributions as p_1, p_2, ..., and p_l. Our algorithm is to find the true distribution by doing pairwise Neyman-Pearson hypothesis testing [58] of these l distributions. We provide the complexity for finding the k anomalous random variables by using the time-invariant mixed measurements in Theorem 4.1.

Theorem 4.1. *Consider time-invariant fixed measurements $Y^j = a^{jT}X^j = a^TX^j$, $1 \le j \le m$, for n random variables $X_1, X_2, ..., X_n$. Algorithms 4.1 and 4.2 correctly identify the k anomalous random variables with high probability, with $O\left(\frac{k\log(n)}{\min_{1\le v,w\le l, v\neq w} C(p_v,p_w)}\right)$ mixed measurements. Here, l is the number of hypotheses, p_v, $1 \le v \le l$, is the output probability distribution for measurements $Y = y$ under hypothesis H_v, and*

$$C(p_v, p_w) = -\min_{0\le\lambda\le 1} \log\left(\int p_v^{\lambda}(y)p_w^{1-\lambda}(y)dy\right)$$

Algorithm 4.1 Likelihood ratio test from deterministic time invariant measurements

Data: observation data $y = [y^1, y^2, ..., y^m]^T$, $l = \binom{n}{k}$

Result: k anomalous random variables

1 For each hypothesis H_v $(1 \le v \le l)$, calculate the likelihood $p_{Y|H_v}(y|H_v)$.

2 Choose the hypothesis with the maximum likelihood.

3 Decide the corresponding k random variables as the anomalous random variables.

is the Chernoff information between two distributions p_v and p_w.

Proof. In Algorithm 4.2, for two probability distributions p_v and p_w, we choose the probability likelihood ratio threshold of the Neyman-Pearson testing in such a way that the error probability decreases with the largest possible error exponent, namely the Chernoff information between p_v and p_w:

$$C(p_v, p_w) = -\min_{0 \le \lambda \le 1} \log\left(\int p_v^{\lambda}(y) p_w^{1-\lambda}(y) dy\right).$$

So overall, the smallest possible error exponent of making an error between any pair of probability distributions is

$$E = \min_{1 \le v,w \le l, v \ne w} C(p_v, p_w).$$

Without loss of generality, we assume that p_1 is the true probability distribution for the observation data $Y = y$. Since the error probability P_e in the Neyman-Pearson testing scales like $P_e \doteq 2^{-mC(p_v,p_w)} \le 2^{-mE}$, by a union bound over the $l-1$ possible pairs (p_1, p_w), the probability that p_1 is not correctly identified as the true probability distribution scales at most as $l \times 2^{-mE}$, where $l = \binom{n}{k}$. Therefore, $\Theta(k \log(n) E^{-1})$

Algorithm 4.2 Hypothesis testing from time-invariant mixed measurements

Data: observation data $y = [y^1, y^2, ..., y^m]^T$

Result: k anomalous random variables

1 For all pairs of distinct probability distributions p_v and p_w $(1 \leq v, w \leq l$ and $v \neq w)$, perform Neyman-Pearson testing for two hypotheses:

- Y^1, Y^2, ..., Y^m follow probability distribution p_v
- Y^1, Y^2, ..., Y^m follow probability distribution p_w

2 Find a certain w^* such that p_{w^*} is the winning probability distribution whenever it is involved in a pairwise hypothesis testing.

3 Declare the k random variables producing p_{w^*} as anomalous random variables.

4 If there is no such w^* in Step 2, then declare a failure in finding the k anomalous random variables.

samplings are enough for identifying the k anomalous samples with high probability.

□

When E grows polynomially with n, this implies a significant reduction in the number of samples needed. If we are allowed to use time-varying non-adaptive sketching functions, we may need fewer samples. In the next subsection, we discuss the performance of time-varying non-adaptive mixed measurements for this problem.

4.3.2 Random time-varying measurements

Inspired by compressed sensing where random measurements often give desirable sparse recovery performance [31, 68], we consider random time-varying measurements. In particular, we assume that each measurement is the inner product between X and one independent realization $a^j = [a_1^j, a_2^j, ..., a_n^j]^T$ of a random sensing vector A at time j. Namely, each observation is given by

$$Y^j = \langle A^j, X^j \rangle = \sum_{i=1}^{n} a_i^j X_i^j, \ 1 \le j \le m,$$

where $a^j = [a_1^j, a_2^j, ..., a_n^j]^T$ is a realization of the random sensing vector A with pdf $p_A(a)$ at time j. We assume that the realizations a^j's of A are independent across different time indices.

We first give the likelihood ratio test algorithm over $\binom{n}{k}$ hypotheses, namely Algorithm 4.3. For the purpose of analyzing the error probability of the likelihood ratio test, we further propose a hypothesis testing algorithm based on pairwise comparison, namely Algorithm 4.4.

The number of samples required to find the abnormal random variables are stated in Theorem 4.2.

Theorem 4.2. *Consider time-varying random measurements $Y^j = y^j$, $1 \le j \le m$, for n random variables X_1, X_2, ..., and X_n. Algorithms 4.3 and 4.4 correctly identify the k anomalous random variables with high probability, in $O\left(\frac{k \log(n)}{\min_{1 \le v,w \le l, v \ne w} IC(p_{A,Y|H_v}, p_{A,Y|H_w})}\right)$ random time-varying measurements. Here, l is the number of hypotheses, $p_{A,Y|H_v}$, $1 \le v \le l$ is the output probability distribution for measurements Y and random sensing*

Algorithm 4.3 Likelihood ratio test from random time-varying measurements

Data: observation and sensing vector at time from 1 to m: $(A^1 = a^1, Y^1 = y^1)$, $(A^2 = a^2, Y^2 = y^2)$, ..., $(A^m = a^m, Y^m = y^m)$

Result: k anomalous random variables

1 For each hypothesis H_v, $1 \le v \le l$, $l = \binom{n}{k}$, calculate the likelihood $p_{A,Y|H_v}(a^j, y^j,\ j = 1, ..., m \mid H_v)$.

2 Choose the hypothesis with the maximum likelihood.

3 Decide the corresponding k random variables as the anomalous random variables.

vectors A under hypothesis H_v, and

$$\begin{aligned} & IC(p_{A,Y|H_v}, p_{A,Y|H_w}) \\ &= -\min_{0\le\lambda\le1} \log\left(\int p_A(a)\left(\int p^{\lambda}_{Y|A,H_v}(y) p^{1-\lambda}_{Y|A,H_w}(y)\, dy\right) da\right) \\ &= -\min_{0\le\lambda\le1} \log\left(E_A\left(\int p^{\lambda}_{Y|A,H_v}(y) p^{1-\lambda}_{Y|A,H_w}(y)\, dy\right)\right) \end{aligned}$$

is the inner conditional Chernoff information *between two hypotheses for measurements Y, conditioned on the probability distribution of time-varying sensing vectors A.*

Proof. In Algorithm 4.4, for two different hypotheses H_v and H_w, we choose the probability likelihood ratio threshold of the Neyman-Pearson testing in a way, such that the hypothesis testing error probability decreases with the largest error exponent, namely the Chernoff information between $p_{A,Y|H_v}$ and $p_{A,Y|H_w}$:

$$IC(p_{A,Y|H_v}, p_{A,Y|H_w}) = -\min_{0\le\lambda\le1} \log\left(\int p^{\lambda}_{A,Y|H_v}(a,y) p^{1-\lambda}_{A,Y|H_w}(a,y) dy\right).$$

Algorithm 4.4 Hypothesis testing from random time-varying measurements

Data: observation and sensing vector at time from 1 to m: $(A^1 = a^1, Y^1 = y^1)$, $(A^2 = a^2, Y^2 = y^2)$, ..., $(A^m = a^m, Y^m = y^m)$

Result: k anomalous random variables

1 For all pairs of hypotheses H_v and H_w ($1 \leq v, w \leq l$ and $v \neq w$, $l = \binom{n}{k}$), perform Neyman-Pearson testing of the following two hypotheses:

- (A^1, Y^1), (A^2, Y^2), ..., (A^m, Y^m) follow the probability distribution $p_{A,Y|H_v}(a, y|H_v)$
- (A^1, Y^1), (A^2, Y^2), ..., (A^m, Y^m) follow probability distribution $p_{A,Y|H_w}(a, y|H_w)$

2 Find a certain w^* such that H_{w^*} is the winning hypothesis, whenever it is involved in a pairwise hypothesis testing.

3 Declare the k random variables producing H_{w^*} as anomalous random variables.

4 If there is no such w^* in Step 2, declare a failure in finding the k anomalous random variables.

Since the random time-varying sensing vectors are independent of random variable X and the hypothesis H_v or H_w, we obtain the following equations:

$$p_{A,Y|H_v}(a, y|H_v) = p(a|H_v)p(y|H_v, a) = p(a)p(y|H_v, a),$$

$$p_{A,Y|H_w}(a, y|H_w) = p(a|H_w)p(y|H_w, a) = p(a)p(y|H_w, a).$$

Then the Chernoff information is simplified to

$$\begin{aligned} & IC(p_{A,Y|H_v}, p_{A,Y|H_w}) \\ &= -\min_{0\le\lambda\le 1} \log\left(\int p_A(a)\left(\int p^{\lambda}_{Y|A,H_v}(y)p^{1-\lambda}_{Y|A,H_w}(y)\,dy\right)da\right) \\ &= -\min_{0\le\lambda\le 1} \log\left(E_A\left(\int p^{\lambda}_{Y|A,H_v}(y)p^{1-\lambda}_{Y|A,H_w}(y)\,dy\right)\right) \end{aligned}$$

Using Holder's inequality, we have

$$IC(p_{A,Y|H_v}, p_{A,Y|H_w}) \ge -\min_{a} \log\left(1 - p_A(a) + p_A(a)e^{-C(p_{Y|A,H_v}, p_{Y|A,H_w})}\right),$$

where

$$C(p_{Y|A,H_v}, p_{Y|A,H_w}) = -\min_{0\le\lambda\le 1} \log\left(\int p^{\lambda}_{Y|A,H_v}(y)p^{1-\lambda}_{Y|A,H_w}(y)dy\right)$$

is the well-known Chernoff information between $p_{Y|A,H_v}$, and $p_{Y|A,H_j}$. So as long as there exist sensing vectors A of a positive probability, such that the regular Chernoff information is positive, then the inner condition Chernoff information $IC(p_{A,Y|H_v}, p_{A,Y|H_w})$ will also be positive.

Overall, the smallest possible error exponent between any pair of hypotheses is

$$E = \min_{1\le v,w\le l, v\ne w} IC(p_{A,Y|H_v}, p_{A,Y|H_w}).$$

Without loss of generality, we assume H_1 is the true hypothesis. Since the error probability P_e in the Neyman-Pearson testing is

$$P_e \doteq 2^{-m(IC(p_{A,Y|H_v}, p_{A,Y|H_w}))} \le 2^{-mE}.$$

By a union bound over the $l-1$ possible pairs (H_1, H_w), the probability that H_1 is not correctly identified as the true hypothesis is upper bounded by $l \times 2^{-mE}$ in terms of

scaling. So $m = \Theta(k \log(n) E^{-1})$ samplings are enough for identifying the k anomalous samples with high probability. When E grows polynomially with n, this implies a significant reduction in the number of required samples. □

4.3.3 Deterministic time-varying measurements

In this subsection, we consider mixed measurements which are allowed to vary over time. However, each sensing vector is predetermined, so that exactly $P(A = a)m$ (assuming that $P(A = a)m$ are integers) measurements use a realized sensing vector a. In contrast, in random time-varying measurements, each sensing vector A is taken randomly, and thus the number of measurements for each random variable X is random. We define the predetermined sensing vector at time j as a^j or A^j in a random variable form.

In deterministic time-varying measurements, we first give the likelihood ratio test algorithm among $l = \binom{n}{k}$ hypotheses in Algorithm 4.5. For the purpose of analyzing the error probability, we consider another hypothesis testing algorithm based on pairwise comparison with deterministic time-varying measurements in Algorithm 4.6.

Theorem 4.3. *Consider time-varying deterministic observations $Y^j = y^j$, $1 \le j \le m$, for n random variables X_1, X_2, ..., and X_n. l is the number of hypotheses for the distribution of the vector $X = [X_1, X_2, ..., X_n]^T$.*

For $\lambda \in [0, 1]$ and two hypotheses H_v and H_w $(1 \le v, w \le l)$, define

$$p_\lambda(y|a, H_v, H_w) = \frac{p^\lambda_{Y|A,H_v}(y|a, H_v) p^{1-\lambda}_{Y|A,H_w}(y|a, H_w)}{\int p^\lambda_{Y|A,H_v}(y|a, H_v) p^{1-\lambda}_{Y|A,H_w}(y|a, H_w) dy},$$

Algorithm 4.5 Likelihood ratio test from deterministic time-varying measurements

Data: observation data $y = [y^1, y^2, ..., y^m]^T$, $y \in \mathbb{R}^{m\times 1}$, deterministic sensing vectors

$[a^1, a^2, ..., a^m]^T \in \mathbb{R}^{m\times n}$

Result: k anomalous random variables

1 For each hypothesis H_v $(1 \le v \le l)$, calculate the likelihood $p_{Y|H_v,A}(y|H_v, a)$.

2 Choose the hypothesis with the maximum likelihood.

3 Declare the corresponding k random variables as the anomalous random variables.

$$Q_{\lambda,v\to w} = E_A\left\{D\left(p_\lambda(y|a, H_v, H_w) \parallel p(y|a, H_v)\right)\right\},$$

$$Q_{\lambda,w\to v} = E_A\left\{D\left(p_\lambda(y|a, H_v, H_w) \parallel p(y|a, H_w)\right)\right\}.$$

Furthermore, we define the outer conditional Chernoff information $OC(p_{Y|A,H_v}, p_{Y|A,H_w})$ *between* H_v *and* H_w*, under deterministic time-varying sensing vector* A*, as*

$$OC(p_{Y|A,H_v}, p_{Y|A,H_w}) = Q_{\lambda,v\to w} = Q_{\lambda,w\to v},$$

where λ *is chosen such that* $Q_{\lambda,v\to w} = Q_{\lambda,w\to v}$.

Then with $O(\frac{k\log(n)}{\min\limits_{1\le v,w\le l, v\ne w} OC(p_{Y|A,H_v}, p_{Y|A,H_w})})$ *random time-varying measurements, with high probability, Algorithms 4.5 and 4.6 correctly identify the* k *anomalous random variables. Here* l *is the number of hypotheses,* $p_{Y|A,H_v}$, $1 \le v \le l$ *is the output probability distribution for observations* Y *under hypothesis* H_v *and sensing vector* A*, and* $OC(p_{Y|A,H_v}, p_{Y|A,H_w})$ *is the outer conditional Chernoff information.*

Algorithm 4.6 Hypothesis testing from deterministic time-varying measurements

Data: observation data $y = [y^1, y^2, ..., y^m]^T$, $y \in \mathbb{R}^{m\times 1}$, deterministic sensing vectors $[a^1, a^2, ..., a^m]^T \in \mathbb{R}^{m\times n}$

Result: k anomalous random variables

1 For all pairs of hypotheses H_v and H_w ($1 \le v, w \le l$ and $v \ne w$), perform Neyman-Pearson testing of the following two hypotheses:

- Y^1, Y^2, ..., Y^m follow the probability distribution $p_{Y|H_v,A}(y|H_v, a)$
- Y^1, Y^2, ..., Y^m follow probability distribution $p_{Y|H_w,A}(y|H_w, a)$

2 Find a certain w^*, such that H_{w^*} is the winning hypothesis, whenever it is involved in a pairwise hypothesis testing.

3 Declare the k random variables producing H_{w^*} as anomalous random variables.

4 If there is no such w^* in Step 2, declare a failure in finding the k anomalous random variables.

Moreover, the outer conditional Chernoff information is equal to

$$\begin{aligned} & OC(p_{Y|A,H_v}, p_{Y|A,H_w}) \\ &= -\min_{0\le\lambda\le 1} \int p_A(a) \log\left(\int p^{\lambda}_{Y|A,H_v}(y) p^{1-\lambda}_{Y|A,H_w}(y)\, dy\right) da \\ &= -\min_{0\le\lambda\le 1} E_A\left(\log\left(\int p^{\lambda}_{Y|A,H_v}(y) p^{1-\lambda}_{Y|A,H_w}(y)\, dy\right)\right). \end{aligned}$$

Proof. In Algorithm 4.6, for two different hypotheses H_v and H_w, we choose the probability likelihood ratio threshold of the Neyman-Pearson testing in a way, such

that the hypothesis testing error probability decreases with the largest error exponent. Now we focus on deriving what this largest error exponent is, under deterministic time-varying measurements. The framework of this proof follows the book of Cover [58, Chap. 11].

For simplicity of presentation, we first consider a special case: there are only two possible sensing vectors a_1 and a_2; and one half of the sensing vectors are a_1 while the other half are a_2. The conclusions can be extended to general distribution $p_A(a)$ on A, in a similar way of reasoning. In addition, we assume that the observation data is over a discrete space χ, which can also be generalized to a continuous space without affecting the conclusion in this theorem. Since we use the probability mass function over a discrete space, we will use upper letters for the probability mass function to distinguish the probability density function over a continuous space in this proof. Suppose we take m measurements in total, our assumption translates to that $\frac{1}{2}m$ measurements are taken from the sensing vector a_1, and $\frac{1}{2}m$ measurements are taken from the sensing vector a_2. Without loss of generality, we consider two hypotheses denoted by H_1 and H_2. Under the sensing vector a_1, we assume that H_1 generates distribution P_1 for observation data; H_2 generates distribution P_2 for observation data. Under a_2, we assume that H_1 generates distribution P_3 for observation data; H_2 generates distribution P_4 for observation data. Please refer to Table 4.2 for the observation distributions under different sensing vectors and different hypotheses.

Suppose that P is the empirical distribution of observation data under the sensing vector a_1, and that P' is the empirical distribution of observation data under

Table 4.2: Hypothesis testing measurement distribution

	H_1	H_2
a_1	P_1	P_2
a_2	P_3	P_4

the sensing vector a_2. Then the Neyman-Pearson testing decides that hypothesis H_1 is true if, for a certain constant T,

$$\frac{1}{2}[D(P\|P_2) - D(P\|P_1)] + \frac{1}{2}[D(P'\|P_4) - D(P'\|P_3)] \geq \frac{1}{n}\log(T),$$

where $D(P\|Q) \triangleq \sum_{y\in\mathcal{X}} P(y)\log\frac{P(y)}{Q(y)}$, which is the relative entropy or Kullback-Leibler distance between two probability mass functions P and Q. By the Sanov's theorem [58], the error exponent of the second kind, namely wrongly deciding "hypothesis H_1 is true" when hypothesis H_2 is actually true, is given by the following optimization problem:

$$\begin{aligned}
&\underset{P,P'}{\text{minimize}} \ \ \frac{1}{2}D(P\|P_2) + \frac{1}{2}D(P'\|P_4)\\
&\text{subject to}\\
&\frac{1}{2}[D(P\|P_2) - D(P\|P_1)] + \frac{1}{2}[D(P'\|P_4) - D(P'\|P_3)] \geq \frac{\log(T)}{n},\\
&\sum_y P(y) = 1,\\
&\sum_y P'(y) = 1.
\end{aligned}$$

Using the Lagrange multiplier method, we try to minimize

$$
\begin{aligned}
&L(P, P', \lambda, v_1, v_2) \\
&= \frac{D(P\|P_2)}{2} + \frac{D(P'\|P_4)}{2} \\
&\quad + \lambda\left(\frac{D(P\|P_2) - D(P\|P_1)}{2} + \frac{D(P'\|P_4) - D(P'\|P_3)}{2} - \frac{\log(T)}{n}\right) \\
&\quad + v_1\left(\sum_y P(y) - 1\right) + v_2\left(\sum_y P'(y) - 1\right).
\end{aligned}
$$

Differentiating with respect to $P(y)$ and $P'(y)$, we get

$$
\begin{aligned}
\frac{1}{2}[\log(\frac{P(y)}{P_2(y)}) + 1 + \lambda\log(\frac{P_1(y)}{P_2(y)})] + v_1 = 0, \\
\frac{1}{2}[\log(\frac{P'(y)}{P_4(y)}) + 1 + \lambda\log(\frac{P_3(y)}{P_4(y)})] + v_2 = 0.
\end{aligned}
$$

From these equations, we can obtain the minimizing P and P',

$$
\begin{aligned}
P = P_\lambda(y|a_1) = \frac{P_1^\lambda(y)P_2^{1-\lambda}(y)}{\sum_{y\epsilon\chi} P_1^\lambda(y)P_2^{1-\lambda}(y)}, \\
P' = P_\lambda(y|a_2) = \frac{P_3^\lambda(y)P_4^{1-\lambda}(y)}{\sum_{y\epsilon\chi} P_3^\lambda(y)P_4^{1-\lambda}(y)},
\end{aligned}
$$

where λ is chosen such that $\frac{1}{2}[D(P\|P_2) - D(P\|P_1)] + \frac{1}{2}[D(P'\|P_4) - D(P'\|P_3)] = \frac{1}{n}\log(T)$.

By symmetry, the error exponent of the second kind, and the error exponent of the first kind are stated as follows respectively:

$$
\begin{aligned}
\tfrac{1}{2}D(P_\lambda(y|a_1) \parallel P_2) + \tfrac{1}{2}D(P_\lambda(y|a_2) \parallel P_4), \\
\tfrac{1}{2}D(P_\lambda(y|a_1) \parallel P_1) + \tfrac{1}{2}D(P_\lambda(y|a_2) \parallel P_3).
\end{aligned}
$$

The first exponent is a non-decreasing function in λ, and the second exponent is a non-increasing function in λ. In fact, the optimal error exponent, which is the

minimum of these two exponents, is achieved when they are equal:

$$\begin{aligned}&\tfrac{1}{2}D(P_\lambda(y|a_1) \parallel P_2) + \tfrac{1}{2}D(P_\lambda(y|a_2) \parallel P_4)\\ =\ &\tfrac{1}{2}D(P_\lambda(y|a_1) \parallel P_1) + \tfrac{1}{2}D(P_\lambda(y|a_2) \parallel P_3).\end{aligned}$$

This finishes the characterization of the optimal error exponent in pairwise hypothesis testing, under deterministic time-varying measurements.

We further prove that this error exponent is equivalent to

$$\begin{aligned}&OC(P_{Y|A,H_v}, P_{Y|A,H_w})\\ =\ &-\min_{0\le\lambda\le1}\int P_A(a)\log\left(\int P^{\lambda}_{Y|A,H_v}(y)P^{1-\lambda}_{Y|A,H_w}(y)\,dy\right)da\\ =\ &-\min_{0\le\lambda\le1}E_A\left(\log\left(\int P^{\lambda}_{Y|A,H_v}(y)P^{1-\lambda}_{Y|A,H_w}(y)\,dy\right)\right).\end{aligned}\tag{4.5}$$

In the proof, we restrict our attention to $H_v = H_1$ and $H_w = H_2$. We will show that the $\lambda \in [0,1]$ that minimizes $E_A\left(\log\left(\int P^{\lambda}_{Y|A,H_v}(y)P^{1-\lambda}_{Y|A,H_w}(y)\,dy\right)\right)$ exactly leads to the following equality:

$$\begin{aligned}&\tfrac{1}{2}D(P_\lambda(y|a_1) \parallel P_2) + \tfrac{1}{2}D(P_\lambda(y|a_2) \parallel P_4)\\ =\ &\tfrac{1}{2}D(P_\lambda(y|a_1) \parallel P_1) + \tfrac{1}{2}D(P_\lambda(y|a_2) \parallel P_3).\end{aligned}\tag{4.6}$$

Moreover, under that minimizer λ, we also obtain the following equality:

$$\begin{aligned}&-E_A\left(\log\left(\int P^{\lambda}_{Y|A,H_v}(y)P^{1-\lambda}_{Y|A,H_w}(y)\,dy\right)\right)\\ =\ &\tfrac{1}{2}D(P_\lambda(y|a_1) \parallel P_1) + \tfrac{1}{2}D(P_\lambda(y|a_2) \parallel P_3).\end{aligned}\tag{4.7}$$

On the one hand, we obtain following equations from (4.6) and the definition

of the relative entropy:

$$\begin{aligned} 0 \overset{(4.6)}{=} & \frac{1}{2}[D(P_\lambda(y|a_1) \parallel P_1) - D(P_\lambda(y|a_1) \parallel P_2)] \\ & + \frac{1}{2}[D(P_\lambda(y|a_2) \parallel P_3) - D(P_\lambda(y|a_2) \parallel P_4)] \\ = & \frac{1}{2}\frac{\sum_y P_1^\lambda(y) P_2^{1-\lambda}(y) \log(\frac{P_2(y)}{P_1(y)})}{\sum_y P_1^\lambda(y) P_2^{1-\lambda}(y)} \\ & + \frac{1}{2}\frac{\sum_y P_3^\lambda(y) P_4^{1-\lambda}(y) \log(\frac{P_4(y)}{P_3(y)})}{\sum_y P_3^\lambda(y) P_4^{1-\lambda}(y)}. \end{aligned} \tag{4.8}$$

Let a particular λ satisfy (4.8). Under this λ, the hypothesis testing error exponent is equal to

$$\frac{1}{2}D(P_\lambda(y|a_1) \parallel P_1) + \frac{1}{2}D(P_\lambda(y|a_2) \parallel P_3),$$

and it is further derived from the definition of the relative entropy as follows:

$$\begin{aligned} & \frac{1}{2}\frac{1}{\sum_y P_1^\lambda(y) P_2^{1-\lambda}(y)} \sum_y \Bigg(P_1^\lambda(y) P_2^{1-\lambda}(y) \\ & \qquad \left[(1-\lambda) \log(\frac{P_2(y)}{P_1(y)}) - \log\Big(\sum_y P_1^\lambda(y) P_2^{1-\lambda}(y) \Big) \right] \Bigg) \\ & + \frac{1}{2}\frac{1}{\sum_y P_3^\lambda(y) P_4^{1-\lambda}(y)} \sum_y \Bigg(P_3^\lambda(y) P_4^{1-\lambda}(y) \\ & \qquad \left[(1-\lambda) \log(\frac{P_4(y)}{P_3(y)}) - \log\Big(\sum_y P_3^\lambda(y) P_4^{1-\lambda}(y) \Big) \right] \Bigg). \end{aligned}$$

Since the first parts of both summations are equal to 0 from (4.8), we have the following equation:

$$\begin{aligned} & \frac{1}{2}D(P_\lambda(y|a_1) \parallel P_1) + \frac{1}{2}D(P_\lambda(y|a_2) \parallel P_3) \\ & = -\frac{1}{2}\log\Big(\sum_y P_1^\lambda(y) P_2^{1-\lambda}(y) \Big) - \frac{1}{2}\log\Big(\sum_y P_3^\lambda(y) P_4^{1-\lambda}(y) \Big), \end{aligned} \tag{4.9}$$

which is just $-E_A\left(\log\left(\int P_{Y|A,H_v}^\lambda(y) P_{Y|A,H_w}^{1-\lambda}(y)\, dy\right)\right)$ under this λ achieving (4.6).

On the other hand, to minimize

$$E_A\left(\log\left(\int P^{\lambda}_{Y|A,H_v}(y)P^{1-\lambda}_{Y|A,H_w}(y)\,dy\right)\right),$$

we set the derivative of (4.9) with respect to λ to 0 as follows:

$$0 \;=\; \frac{1}{2}\frac{\sum_y P_1^{\lambda}(y)P_2^{1-\lambda}(y)\log(\frac{P_2(y)}{P_1(y)})}{\sum_y P_1^{\lambda}(y)P_2^{1-\lambda}(y)} + \frac{1}{2}\frac{\sum_y P_3^{\lambda}(y)P_4^{1-\lambda}(y)\log(\frac{P_4(y)}{P_3(y)})}{\sum_y P_3^{\lambda}(y)P_4^{1-\lambda}(y)}.$$

It is noteworthy that (4.10) is the same as (4.8). Let us denote a minimizer $\lambda_{min} = \arg\min_{0\le\lambda\le 1} E_A\left(\log\left(\int P^{\lambda}_{Y|A,H_v}(y)P^{1-\lambda}_{Y|A,H_w}(y)\,dy\right)\right)$. Then, when $\lambda = \lambda_{min}$, (4.8) is satisfied. Furthermore, for $\lambda = \lambda_{min}$, we obtain (4.9). It states that for $\lambda = \lambda_{min}$, the following equation is satisfied:

$$-E_A\left(\log\left(\int P^{\lambda}_{Y|A,H_v}(y)P^{1-\lambda}_{Y|A,H_v}(y)\,dy\right)\right) = \frac{1}{2}D(P_\lambda(y|a_1)\parallel P_1) + \frac{1}{2}D(P_\lambda(y|a_2)\parallel P_3). \tag{4.10}$$

Therefore, we conclude the equivalence of the two different definitions of outer Chernoff information.

Overall, the smallest possible error exponent between any pair of hypotheses is

$$E = \min_{1\le v,w\le l, v\ne w} OC(P_{Y|A,H_v}, P_{Y|A,H_w}).$$

Without loss of generality, we assume H_1 is the true hypothesis. Since the error probability P_e in the Neyman-Pearson testing is

$$P_e \triangleq 2^{-mOC(P_{Y|A,H_v},P_{Y|A,H_w})} \le 2^{-mE}.$$

By a union bound over the $l-1$ possible pairs (H_1, H_w), the probability that H_1 is not correctly identified as the true hypothesis is upper bounded by $l\times 2^{-mE}$ in terms of

scaling, where $l = \binom{n}{k}$. Therefore, $m = \Theta(k \log(n) E^{-1})$ samplings are enough for identifying the k anomalous samples with high probability. When E grows polynomially with n, this implies a significant reduction in the number of samples needed. □

4.4 Examples of Compressed Hypothesis Testing

In this section, we provide simple examples in which smaller error probability can be achieved in hypothesis testing through mixed observations than the traditional individual measurement based approach, under the same number of measurements. In particular, we consider Gaussian distributions in our examples. We also introduce the method to obtain the optimal sensing vectors in a given problem.

4.4.1 Example 1: two Gaussian random variables

In this example, we consider $n = 2$, and $k = 1$. We group the two random variables X_1 and X_2 in a random vector $[X_1, X_2]^T$. Suppose that there are two hypotheses for a 2-dimensional random vector $[X_1, X_2]^T$, where X_1 and X_2 are independent:

- H_1: $X_1 \sim \mathcal{N}(A, \sigma^2)$ and $X_2 \sim \mathcal{N}(B, \sigma^2)$,
- H_2: $X_1 \sim \mathcal{N}(B, \sigma^2)$ and $X_2 \sim \mathcal{N}(A, \sigma^2)$.

Here A and B are two distinct constants, and σ^2 is the variance of the two Gaussian random variables. At each time index, only one observation is allowed, and the observation is restricted to a linear mixing of X_1 and X_2. Namely

$$Y^j = a_1 X_1 + a_2 X_2.$$

We assume that the sensing vector $[a_1, a_2]^T$ does not change over time.

Clearly, when $a_1 \neq 0$ and $a_2 = 0$, the sensing vector reduces to a separate observation of X_1; and when $a_1 = 0$ and $a_2 \neq 0$, it reduces to a separate observation of X_2. In both these two cases, the observation follows distribution $\mathcal{N}(A, \sigma^2)$ for one hypothesis, and follows distribution $\mathcal{N}(B, \sigma^2)$ for the other hypothesis. The Chernoff information between these two distributions are

$$C(\mathcal{N}(A, \sigma^2), \mathcal{N}(B, \sigma^2)) = \frac{(A-B)^2}{8\sigma^2}.$$

When hypothesis H_1 holds, the observation Y^j follows Gaussian distribution $\mathcal{N}(a_1 A + a_2 B, (a_1^2 + a_2^2)\sigma^2)$. Similarly, when hypothesis H_2 holds, the observation Y^j follows Gaussian distribution $\mathcal{N}(a_1 B + a_2 A, (a_1^2 + a_2^2)\sigma^2)$. The Chernoff information between these two Gaussian distributions $\mathcal{N}(a_1 A + a_2 B, (a_1^2 + a_2^2)\sigma^2)$ and $\mathcal{N}(a_1 B + a_2 A, (a_1^2 + a_2^2)\sigma^2)$ is given by

$$\frac{[(a_1 A + a_2 B) - (a_1 B + a_2 A)]^2}{8(a_1^2 + a_2^2)\sigma^2} = \frac{[(a_1 - a_2)^2 (A-B)^2]}{8(a_1^2 + a_2^2)\sigma^2} \leq \frac{2(A-B)^2}{8\sigma^2},$$

where the last inequality follows from Cauchy-Schwarz inequality, and takes equality when $a_1 = -a_2$.

Compared with the Chernoff information for separate observations of X_1 or X_2, the linear mixing of X_1 and X_2 doubles the Chernoff information. This shows that linear mixed observations can offer strict improvement in terms of reducing the error probability in hypothesis testing, by increasing the error exponent.

4.4.2 Example 2: Gaussian random variables with different means

In this example, we consider the mixed observations for two hypotheses of Gaussian random vectors. In general, suppose that there are two hypotheses for an n-dimensional random vector $[X_1, X_2, ..., X_n]^T$,

- H_1: $[X_1, X_2, ..., X_n]$ follow jointly Gaussian distributions $\mathcal{N}(\mu_1, \Sigma_1)$,
- H_2: $[X_1, X_2, ..., X_n]$ follow jointly Gaussian distributions $\mathcal{N}(\mu_2, \Sigma_2)$.

Here Σ_1 and Σ_2 are both $n \times n$ covariance matrices.

At each time instant, only one observation is allowed, and the observation is restricted to a time-invariant sensing vector. Namely

$$Y^j = \langle A, X^j \rangle,$$

where $A \in \mathbb{R}^{n \times 1}$.

Under these conditions, the observation follows distribution $\mathcal{N}(A^T\mu_1, A^T\Sigma_1 A)$ for hypothesis H_1, and follows distribution $\mathcal{N}(A^T\mu_2, A^T\Sigma_2 A)$ for the other hypothesis H_2. We would like to choose a sensing vector A which maximizes the Chernoff information between the two possible univariate Gaussian distributions, namely

$$\max_A C(\mathcal{N}(A^T\mu_1, A^T\Sigma_1 A), \mathcal{N}(A^T\mu_2, A^T\Sigma_2 A)).$$

In fact, the Chernoff information between these two distributions [145] is

$$\begin{aligned} & C(\mathcal{N}(A^T\mu_1, A^T\Sigma_1 A), \mathcal{N}(A^T\mu_2, A^T\Sigma_2 A)) \\ = & \max_{0\le\alpha\le 1}\left[\frac{1}{2}\log\left(\frac{A^T(\alpha\Sigma_1+(1-\alpha)\Sigma_2)A}{(A^T\Sigma_1 A)^\alpha (A^T\Sigma_2 A)^{1-\alpha}}\right)\right. \\ & \left. + \frac{\alpha(1-\alpha)(A^T(\mu_1-\mu_2))^2}{2A^T(\alpha\Sigma_1+(1-\alpha)\Sigma_2)A}\right]. \end{aligned}$$

We first look at the special case when $\Sigma = \Sigma_1 = \Sigma_2$. Under this condition, the maximum Chernoff information is given by

$$\max_A \max_{0\le\alpha\le 1} \frac{\alpha(1-\alpha)[A^T(\mu_1-\mu_2)]^2}{2A^T\Sigma A}.$$

Taking $A' = \Sigma^{\frac{1}{2}}A$, then this reduces to

$$\max_{A'} \max_{0\le\alpha\le 1} \frac{\alpha(1-\alpha)[(A')^T\Sigma^{-\frac{1}{2}}(\mu_1-\mu_2)]^2}{2(A')^T A'}.$$

From Cauchy-Schwarz inequality, it is easy to see that the optimal $\alpha = \frac{1}{2}$, $A' = \Sigma^{-\frac{1}{2}}(\mu_1-\mu_2)$, and $A = \Sigma^{-1}(\mu_1-\mu_2)$. Under these conditions, the maximum Chernoff information is given by

$$\frac{1}{8}(\mu_1-\mu_2)^T\Sigma^{-1}(\mu_1-\mu_2).$$

Note that in general, $A' = \Sigma^{-\frac{1}{2}}(\mu_1-\mu_2)$ is not a separate observation of a certain individual random variable, but rather a linear mixing of the n random variables.

4.4.3 Example 3: Gaussian random variables with different variances

In this example, we look at the mixed observations for Gaussian random variables with different variances. Consider the same setting in Example 2, except that we now look at the special case when $\mu = \mu_1 = \mu_2$. We will study the optimal sensing

vector under this scenario. Then the Chernoff information becomes

$$\begin{aligned} & C(\mathcal{N}(A^T\mu, A^T\Sigma_1 A), \mathcal{N}(A^T\mu, A^T\Sigma_2 A)) \\ = & \max_{0\le\alpha\le 1} \frac{1}{2}\log\left(\frac{A^T(\alpha\Sigma_1 + (1-\alpha)\Sigma_2)A}{(A^T\Sigma_1 A)^\alpha (A^T\Sigma_2 A)^{1-\alpha}}\right). \end{aligned}$$

To find the optimal sensing vector A, we are solving this optimization problem

$$\max_A \max_{0\le\alpha\le 1} \frac{1}{2}\log\left(\frac{A^T(\alpha\Sigma_1 + (1-\alpha)\Sigma_2)A}{(A^T\Sigma_1 A)^\alpha (A^T\Sigma_2 A)^{1-\alpha}}\right).$$

For a certain A, we define

$$B = \frac{\max\left(A^T\Sigma_1 A, A^T\Sigma_2 A\right)}{\min\left(A^T\Sigma_1 A, A^T\Sigma_2 A\right)}.$$

Note that $B \ge 1$. By symmetry over α and $1-\alpha$, maximizing the Chernoff information can always be reduced to

$$\max_{B\ge 1} \max_{0\le\alpha\le 1} \frac{1}{2}\log\left(\frac{\alpha + (1-\alpha)B}{B^{1-\alpha}}\right). \tag{4.11}$$

The optimal α is obtained as follows by finding the point which makes the first order differential equation to zero:

$$\alpha = \frac{-(B-1) + B\log(B)}{(B-1)\log(B)}.$$

By plugging the obtained optimal α to (4.11), we obtain the following optimization problem:

$$\max_{B\ge 1} \frac{1}{2}\left\{-1 + \frac{B}{B-1}\log(B) + \log\left(\frac{B-1}{B\log(B)}\right)\right\}. \tag{4.12}$$

We note that the objective function is an increasing function in B, when $B \ge 1$.

Lemma 4.4. *The following optimization problem*

$$\max_{0\leq\alpha\leq 1} \frac{1}{2}\log\left(\frac{\alpha+(1-\alpha)B}{B^{1-\alpha}}\right),$$

is an increasing function in $B \geq 1$.

Proof. We only need to show that for any $\alpha \in [0,1]$, $\left(\frac{\alpha+(1-\alpha)B}{B^{1-\alpha}}\right)$ is an increasing function in $B \geq 1$. In fact, the derivative of it with respect to α is

$$\alpha(1-\alpha)(B^{\alpha-1}-B^{\alpha-2}) \geq 0.$$

Then the conclusion of this lemma immediately follows. □

This means we need to maximize B, in order to maximize the Chernoff information. Hence, to find the optimal A maximizing B, we solve the following two optimization problems:

$$\max_{A} A^T\Sigma_1 A \qquad \text{subject to} \quad A^T\Sigma_2 A \leq 1, \tag{4.13}$$

and

$$\max_{A} A^T\Sigma_2 A \qquad \text{subject to} \quad A^T\Sigma_1 A \leq 1. \tag{4.14}$$

Then the maximum of the two optimal objective values is equal to the optimal objective value of optimizing B, and the corresponding A is the optimal sensing vector maximizing the Chernoff information. These two optimization problems are not convex optimization programs, however, they still admit zero duality gap from the S-procedure, and can be efficiently solved [20]. In fact, they are respectively equivalent

to the following two semidefinite programming optimization problems:

$$\begin{aligned} \underset{\gamma,\lambda}{\text{minimize}} \quad & -\gamma \\ \text{subject to} \quad & \lambda \geq 0 \\ & \begin{pmatrix} -\Sigma_1 + \lambda\Sigma_2 & 0 \\ 0 & -\lambda - \gamma \end{pmatrix} \geq 0, \end{aligned} \tag{4.15}$$

and

$$\begin{aligned} \underset{\gamma,\lambda}{\text{minimize}} \quad & -\gamma \\ \text{subject to} \quad & \lambda \geq 0 \\ & \begin{pmatrix} -\Sigma_2 + \lambda\Sigma_1 & 0 \\ 0 & -\lambda - \gamma \end{pmatrix} \geq 0. \end{aligned} \tag{4.16}$$

Thus they can be efficiently solved.

For example, when Σ_1 and Σ_2 are given as follows:

$$\Sigma_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 100 \end{bmatrix}, \qquad \Sigma_2 = \begin{bmatrix} 100 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

we obtain $[0, 0, -1]^T$ for the optimal sensing vector A. This is because the diagonal elements in the covariance matrices Σ_1 and Σ_2 represent the variance of random variables $[X_1, X_2, X_3]^T$, and the biggest difference in variance is shown in random variable X_1 or X_3. Therefore, checking the random variable X_3 (or X_1) through realizations is an optimal way to figure out whether the random variables follows Σ_1 or Σ_2. On the other hand, if the random variables are dependent, and Σ_1 and Σ_2 are

given as follows:

$$\Sigma_1 = \begin{bmatrix} 1 & 0.5 & 0.5 \\ 0.5 & 1 & 0.5 \\ 0.5 & 0.5 & 100 \end{bmatrix}, \qquad \Sigma_2 = \begin{bmatrix} 100 & 0.5 & 0.5 \\ 0.5 & 1 & 0.5 \\ 0.5 & 0.5 & 1 \end{bmatrix},$$

then, we obtain $[0.4463, 0.0022, -0.8949]^T$ for the optimal sensing vector A which maximizes the Chernoff information.

4.4.4 Example 4: $k = 1$ anomalous random variable among $n = 7$ random variables

Consider another example, where $k = 1$ and $n = 7$. $n - k = 6$ random variables follow the distribution $\mathcal{N}(0, 1)$; and the other random variable follows distribution $\mathcal{N}(0, \sigma^2)$, where $\sigma^2 \neq 1$. We assume that all random variables X_1, X_2,..., and X_7 are independent. So overall, there are 7 hypotheses:

- H_1: $(X_1, X_2, ..., X_7) \sim (\mathcal{N}(0, \sigma^2), \mathcal{N}(0, 1), ..., \mathcal{N}(0, 1))$,
- H_2: $(X_1, X_2, ..., X_7) \sim (\mathcal{N}(0, 1), \mathcal{N}(0, \sigma^2), ..., \mathcal{N}(0, 1))$,
- .....
- H_7: $(X_1, X_2, ..., X_7) \sim (\mathcal{N}(0, 1), \mathcal{N}(0, 1), ..., \mathcal{N}(0, \sigma^2))$.

We first assume that separate observations of these 7 random variables are obtained. For any pair of hypotheses H_v and H_w, the probability distributions for the output are respectively $\mathcal{N}(0, \sigma^2)$ and $\mathcal{N}(0, 1)$, when X_v is observed; the probability distributions for the output are respectively $\mathcal{N}(0, 1)$ and $\mathcal{N}(0, \sigma^2)$, when X_w is

observed. From (4.5), the Chernoff information between H_v and H_w is given by

$$\begin{aligned}
&OC(p_{Y|A,H_v}, p_{Y|A,H_w}) \\
&= -\min_{0\le\lambda\le 1} E_A\left(\log\left(\int p^{\lambda}_{Y|A,H_v}(y)p^{1-\lambda}_{Y|A,H_v}(y)\,dy\right)\right) \\
&= -\min_{0\le\lambda\le 1}\left[\frac{1}{7}\left(\log\left(\int p^{\lambda}_{\mathcal{N}(0,1)}(y)p^{1-\lambda}_{\mathcal{N}(0,\sigma^2)}(y)\,dy\right)\right)\right. \\
&\quad \left.+\frac{1}{7}\left(\log\left(\int p^{\lambda}_{\mathcal{N}(0,\sigma^2)}(y)p^{1-\lambda}_{\mathcal{N}(0,1)}(y)\,dy\right)\right)\right].
\end{aligned}$$

Optimizing over λ, we obtain the optimal $\lambda = \frac{1}{2}$, and that

$$OC(p_{Y|A,H_v}, p_{Y|A,H_w}) = \frac{1}{7}\log\left(\frac{B+1}{2B^{\frac{1}{2}}}\right), \tag{4.17}$$

where $B = \frac{\max(\sigma^2,1)}{\min(\sigma^2,1)}$.

Now we consider using the parity check matrix of $(7,4)$ Hamming codes to obtain measurements as follows:

$$\begin{bmatrix} a^{1T} \\ a^{2T} \\ a^{3T} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix}.$$

We use the row vectors of the parity check matrix as sensing vectors for deterministic time varying measurements. For example, the i-th row vector is used for the j-th mixed measurement, where $i = (j \bmod 3) + 1$. For any pair of hypotheses H_v and H_w, there is always a sensing vector among 3 sensing vectors which measures one and only one of X_v and X_w. Without loss of generality, we assume that that a sensing vector measures X_v but not X_w. Suppose H_v is true, then the output probability distribution for that measurement is $\mathcal{N}(0, \sigma^2 + s(A))$; otherwise when H_w is true, the output probability distribution is given by $\mathcal{N}(0, 1 + s(A))$, where $1 + s(A)$ is the

number of ones in the sensing vector A. The Chernoff information between H_v and H_w is bounded by

$$\begin{aligned}
&OC(p_{Y|A,H_v}, p_{Y|A,H_w}) \\
&= -\min_{0\le\lambda\le 1} E_A\left(\log\left(\int p^{\lambda}_{Y|A,H_v}(y)p^{1-\lambda}_{Y|A,H_v}(y)\,dy\right)\right) \\
&\ge -\min_{0\le\lambda\le 1} \frac{1}{3}\log\left(\int p^{\lambda}_{\mathcal{N}(0,\sigma^2+s(A))}(y)p^{1-\lambda}_{\mathcal{N}(0,1+s(A))}(y)\,dy\right),
\end{aligned}$$

where the final inequality is obtained by considering only one sensing vector among 3 sensing vectors. From (4.12), this lower bound is given by

$$\frac{1}{6}\left\{-1+\frac{B}{B-1}\log(B)+\log\left(\frac{B-1}{B\log(B)}\right)\right\},$$

where

$$B=\frac{\max\left(\sigma^2+s(A),1+s(A)\right)}{\min\left(\sigma^2+s(A),1+s(A)\right)}.$$

Simply taking $\lambda=\frac{1}{2}$, we obtain another lower bound of $OC(p_{Y|A,H_v}, p_{Y|A,H_w})$ as follows:

$$\begin{aligned}
&-\min_{0\le\lambda\le 1} \frac{1}{3}\log\left(\int p^{\lambda}_{\mathcal{N}(0,\sigma^2+s(A))}(y)p^{1-\lambda}_{\mathcal{N}(0,1+s(A))}(y)\,dy\right) \\
&\ge \begin{cases} \frac{1}{3}\times\frac{1}{2}\log\left(\frac{\sigma^2+s(A)+1}{2\sqrt{\sigma+s(A)}}\right), & \text{if } \sigma^2\ge 1, \\ \frac{1}{3}\times\frac{1}{2}\log\left(\frac{s(A)+2}{2\sqrt{1+s(A)}}\right), & \text{if } \sigma^2<1. \end{cases}
\end{aligned}$$

When $\sigma^2 \gg 1$, for separate observations,

$$OC(p_{Y|A,H_v}, p_{Y|A,H_w}) = \frac{1}{7}\log(\frac{\sigma^2+1}{2\sigma}) \sim \frac{1}{7}\log(\frac{\sigma}{2});$$

while for measurements through the parity-check matrix of Hamming codes, a lower

bound of $OC(p_{Y|A,H_v}, p_{Y|A,H_w})$ asymptotically satisfies

$$OC(p_{Y|A,H_v}, p_{Y|A,H_w}) = \frac{1}{6}\log\left(\frac{\sigma^2 + s(A) + 1}{2\sqrt{\sigma^2 + s(A)}}\right) \sim \frac{1}{6}\log\frac{\sigma}{2}.$$

So in the end, the minimum Chernoff information between any pair of hypotheses, under mixed measurements using Hamming codes as sensing vectors, is bigger than the Chernoff information obtained using separate observations. This means that mixed observations can offer strict improvement in the error exponent of hypothesis testing problems.

4.5 Characterization of Optimal Sensing Vector Maximizing the Error Exponent

In this section, we derive a characterization of the optimal deterministic time-varying measurements which maximize the error exponent of hypothesis testing. We further explicitly design the optimal measurements for some simple examples. We begin with the following lemma about the error exponent of hypothesis testing.

Lemma 4.5. *Suppose that there are overall* $l = \binom{n}{k}$ *hypotheses. For any fixed* k *and* n*, the error exponent of the error probability of hypothesis testing is given by*

$$E = \min_{1\le v,w\le l, v\neq w} OC(p_{Y|A,H_v}, p_{Y|A,H_w}).$$

Proof. We first give an upper bound on the error probability of hypothesis testing. Without loss of generality, we assume H_1 is the true hypothesis. The error probability

P_e in the Neyman-Pearson testing is stated as follows:

$$P_e = 2^{-mOC(p_{Y|A,H_v}, p_{Y|A,H_w})} \leq 2^{-mE}.$$

By a union bound over the $l-1$ possible pairs (H_1, H_w), the probability that H_1 is not correctly identified as the true hypothesis is upper bounded by $l \times 2^{-mE}$ in terms of scaling.

Now we give a lower bound on the error probability of hypothesis testing. Without loss of generality, we assume that E is achieved between the hypothesis H_1 and the hypothesis H_2, namely,

$$E = OC(p_{Y|A,H_1}, p_{Y|A,H_2}).$$

Suppose that we are given the prior information that either hypothesis H_1 or H_2 is true. Knowing this prior information will not increase the error probability. Under this prior information, the error probability behaves asymptotically as 2^{-mE} as $m \to \infty$. This shows that the error exponent of hypothesis testing is no bigger than E. □

The following theorem gives a simple characterization of the optimal probability density function $p_A(a)$ for the sensing vector. This enables us to explicitly find the optimal sensing vectors, under certain special cases of Gaussian random variables.

Theorem 4.6. *In order to maximize the error exponent in hypothesis testing, the optimal sensing vectors have a distribution $p_A^*(a)$ which maximizes the minimum of the pairwise outer Chernoff information between different hypotheses:*

$$p_A^*(a) = arg \max_{p_A(a)} \min_{1 \leq v,w \leq l, v \neq w} OC(p_{Y|A,H_v}, p_{Y|A,H_w}).$$

When $k = 1$ and the n random variables of interest are independent Gaussian random variables with the same variances, the optimal $p_A^*(a)$ admits a discrete probability distribution:

$$p_A^*(a) = \sum_{\sigma \text{ as a permutation}} \frac{1}{n!}\delta(a - \sigma(a^*)),$$

where a^* is a constant n-dimensional vector such that

$$a^* = arg\max_a \sum_{1\le v,w\le l, v\neq w} C(p_{Y|A,H_v}, p_{Y|A,H_w}).$$

Proof. The first statement follows from Lemma 4.5. So we only need to prove the optimal sensing vectors for Gaussian random variables with the same variance under $k = 1$. We let $l = \binom{n}{k}$. For any $p_A(a)$, we have

$$\begin{aligned}
&\min_{1\le v,w\le l, v\neq w} OC(p_{Y|A,H_v}, p_{Y|A,H_w}) \\
\le\ & \frac{1}{\binom{n}{2}} \sum_{1\le v,w\le l, v\neq w} OC(p_{Y|A,H_v}, p_{Y|A,H_w}) \\
\le\ & \frac{1}{\binom{n}{2}} \sum_{1\le v,w\le l, v\neq w} \int p_A(a) C(p_{Y|A,H_v}, p_{Y|A,H_w})\, da \\
=\ & \frac{1}{\binom{n}{2}} \int p_A(a) \sum_{1\le v,w\le l, v\neq w} C(p_{Y|A,H_v}, p_{Y|A,H_w})\, da \\
\le\ & \frac{1}{\binom{n}{2}} \sum_{1\le v,w\le l, v\neq w} C(p_{Y|A^*,H_v}, p_{Y|A^*,H_w}),
\end{aligned}$$

where the second inequality follows from the fact that for any functions $f_i(\lambda)$, $i = 1, ..., l$, and for any λ in the intersection domain of all the $f_i(\lambda)$'s, $\min_\lambda(f_1(\lambda) + f_2(\lambda) + ... + f_l(\lambda)) \ge \min_\lambda f_1(\lambda) + \min_\lambda f_2(\lambda) + ... + \min_\lambda f_l(\lambda)$, and A^* is the same expression for a^* in the form of random variable.

On the other hand, for two Gaussian distributions with the same variances, the optimal λ in (4.5) is always equal to $\frac{1}{2}$, no matter what $p_A(a)$ is chosen. By symmetry, when

$$p_A^*(a) = \sum_{\sigma \text{ as a permutation}} \frac{1}{n!}\delta(a - \sigma(a^*)),$$

and $\lambda = \frac{1}{2}$, for any two different hypotheses H_v and H_w,

$$\begin{aligned}
& OC(p_{Y|A,H_v}, p_{Y|A,H_w}) \\
= & \int p_A^*(a) C(p_{Y|A,H_v}, p_{Y|A,H_w})\, da \\
= & \frac{1}{\binom{n}{2}} \int p_A^*(a) \sum_{1\le v,w\le l, v\neq w} C(p_{Y|A,H_v}, p_{Y|A,H_w})\, da \\
= & \int p_A^*(a) \frac{1}{\binom{n}{2}} \sum_{1\le v,w\le l, v\neq w} C(p_{Y|A^*,H_v}, p_{Y|A^*,H_w})\, da \\
= & \frac{1}{\binom{n}{2}} \sum_{1\le v,w\le l, v\neq w} C(p_{Y|A^*,H_v}, p_{Y|A^*,H_w}),
\end{aligned}$$

where the first equality is from the fact that $\lambda = \frac{1}{2}$ is the common maximizer for Chernoff information, the second equality is from the permutation symmetry of $p_A^*(a)$, the third equality is again from the generation of $p_A^*(a)$ from permutations of a^*, and the last equality follows from $\int p_A^*(a)\, da = 1$.

This means that under $p_A^*(a)$,

$$\begin{aligned}
& \min_{1\le v,w\le l, v\neq w} OC(p_{Y|A,H_v}, p_{Y|A,H_w}) \\
= & \frac{1}{\binom{n}{2}} \sum_{1\le v,w\le l, v\neq w} C(p_{Y|A^*,H_v}, p_{Y|A^*,H_w}).
\end{aligned}$$

This further implies that the upper bound on $\min_{1\le v,w\le l, v\neq w} OC(p_{Y|A,H_v}, p_{Y|A,H_w})$ is achieved, and we can conclude that $p_A^*(a)$ is the optimal distribution. □

We can use Theorem 4.6 to calculate explicitly the optimal sensing vectors for n independent Gaussian random variables of the same variance σ^2, among which $k = 1$ random variable has a mean μ_1 and the mean of the other $(n-1)$ random variables is μ_2. To obtain the optimal measurements maximizing the error exponent E, we first need to calculate a constant vector a^* such that

$$a^* = \arg\max_{a} \sum_{1\le v,w\le l, v\neq w} C(p_{Y|A,H_v}, p_{Y|A,H_w}).$$

After a simple calculation, a^* is the optimal solution to

$$\begin{aligned} \underset{a}{\text{maximize}} \quad & \frac{1}{\binom{n}{2}} \sum_{1\le i,j\le n, i\neq j} (a_i - a_j)^2 \\ \text{subject to} \quad & \sum_{i=1}^{n} a_i^2 \le 1, \end{aligned} \tag{4.18}$$

where a_i represents the i-th element of vector a, and the corresponding optimal error exponent is

$$\frac{1}{\binom{n}{2}} \sum_{1\le i,j\le n, i\neq j} \frac{(a_i^* - a_j^*)^2(\mu_1-\mu_2)^2}{8\sigma^2 \sum_{i=1}^{n} (a_i^*)^2}.$$

This optimization problem is not a convex optimization program, however, it still admits zero duality gap from the S-procedure, and can be efficiently solved [20]. In fact, we get the optimal solution $a^* = [\frac{1}{\sqrt{2}}, \frac{-1}{\sqrt{2}}, 0, ..., 0]^T$. Then an optimal distribution $p_A^*(a)$ for the sensing vector is

$$p_A^*(a) = \sum_{\sigma \text{ as a permutation}} \frac{1}{n!} \delta(a - \sigma(a^*)).$$

Namely, this optimal sensing vector is to uniformly choose two random variables, say X_1 and X_2, and take their weighted sum $\frac{1}{\sqrt{2}}X_1 - \frac{1}{\sqrt{2}}X_2$. Correspondingly, the optimal

error exponent is

$$\frac{(\mu_1-\mu_2)^2}{8\sigma^2}\frac{n}{\binom{n}{2}}=\frac{(\mu_1-\mu_2)^2}{4\sigma^2(n-1)}.$$

In contrast, if we perform separate observations of n random variables individually, the error exponent will be

$$\frac{(\mu_1-\mu_2)^2}{8\sigma^2}\frac{n-1}{\binom{n}{2}}=\frac{(\mu_1-\mu_2)^2}{4\sigma^2 n}.$$

In fact, linear mixed observations increase the error exponent by $\frac{n}{n-1}$ times. When n is small, the improvement is significant. For example, when $n=2$, the error exponent is doubled.

From this example of Gaussian random variables with different means, we can make some interesting observations. On the one hand, quite surprisingly, separate observations of random variables are *not* optimal in maximizing the error exponent. On the other hand, in the optimal measurement scheme, each measurement takes a linear mixing of only two random variables, instead of mixing all the random variables. It is interesting to see whether these observations hold for other types of random variables.

4.6 Efficient Algorithms for Hypothesis Testing from Mixed Observations

In Section 4.3, we introduce the likelihood ratio test algorithms for three different types of mixed observations. Even though the likelihood ratio test is the optimal test achieving the smallest hypothesis testing error probability, conducting the likelihood ratio test over $\binom{n}{k}$ hypotheses is computationally challenging. Especially, when

n and k are large, finding k abnormal random variables out of n random variables is almost impossible by using the likelihood ratio test. To overcome this drawback, we further design efficient algorithms to find k abnormal random variables among n random variables for large n and k by using Least Absolute Shrinkage and Selection Operator (LASSO) algorithm [170], and Message Passing (MP) algorithm [129].

4.6.1 LASSO based hypothesis testing algorithm

We propose to use the LASSO algorithm to detect k anomalous random variables when they have different means from the other $n-k$ random variables. Without loss of generality, we assume that each of the abnormal random variables has a non-zero mean, while each of the other $(n-k)$ random variables has mean 0.

LASSO, also known as Basis Pursuit denoising [44], is a well-known sparse regression tool in statistics, and has been successfully applied to various fields such as signal processing [43], machine learning [162], and control system [117]. For a sensing matrix $A \in R^{m \times n}$, and observation $y \in R^m$, the LASSO optimization formulation is given as follows:

$$\underset{x}{\text{minimize}} \ \frac{1}{2}||y - Ax||_2^2 + \lambda||x||_1, \tag{4.19}$$

where x is a variable, and λ is a parameter for the penalty term, ℓ_1 norm of x, which makes more elements of x driven to zero as λ increases.

We use the LASSO formulation (4.19) to obtain a solution $\hat{x}$, by taking $y = [Y^1, ..., Y^m]^T$ and A as an $m \times n$ matrix with its j-th row equal to the sensing vector A^j. Since we are interested in finding k abnormal random variables, we solve (4.19)

and select k largest elements of $\hat{x}$ in amplitude. We decide the corresponding k random variables as the k abnormal random variables.

4.6.2 MP based hypothesis testing algorithm

We further design a message passing method to discover general abnormal random variables, even if the abnormal random variables have the same mean as the regular random variables. Our message passing algorithm uses *bipartite graph* to perform statistical inference. We remark that message passing algorithms have been successfully applied to decoding for error correcting code [129], including Low Density Parity Check (LDPC) code.

Let us denote the j-th linear mixed function by ${a^j}^T X$, where X is a random variable vector, and a^j is the j-th sensing vector for a linear mixed observation. The observation random variable at time j, i.e. Y^j, is simply represented as ${a^j}^T X^j$ as in (4.2). Note that over time indices j's, the random variable vector X takes independent realizations. Now we define a bipartite factor graph as follows. We will represent a random variable X_i using a *variable node* on the left, and represent an observation Y^j as a *check node* on the right. Variable nodes and check nodes appear on two sides of the bipartite factor graph. A line is drawn between a node X_i and a node Y^j if and only if that random variable X_i is involved in the observation Y^j. We call the observation Y^j linked with X_i as a *neighbor* of X_i. We will denote the set of neighbors of X_i by $N(X_i)$. Similarly, the set of random variable X_i's linked with a random variable Y^j is denoted by $N(Y^j)$. Fig. 4.1(a) is an example of the factor

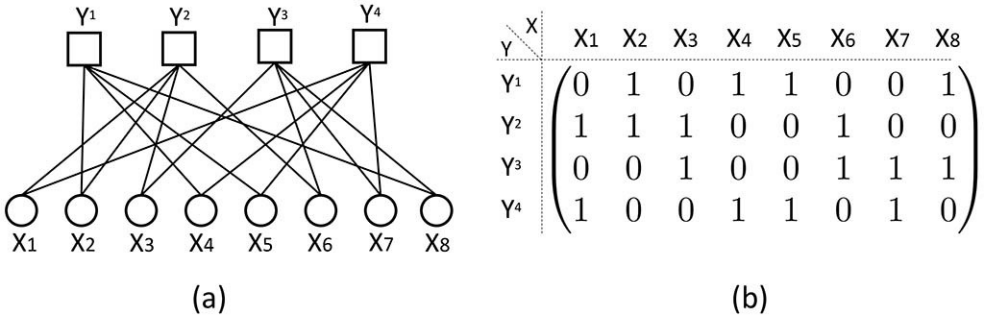


Figure 4.1: Illustration of a factor graph (a) from a matrix (b). A random variable X_i and Y^j are considered as a variable node and a check node in the graph respectively.

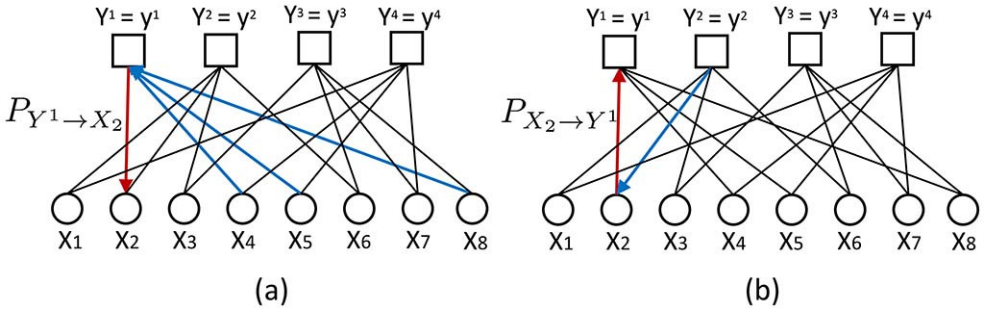


Figure 4.2: (a) Message sent from a check node to a variable node. The message sent from a check node Y^1 to a variable node X_2 (red arrow) is expressed as the probability $P_{Y^1 \to X_2}(X_2 \text{ is abnormal} \mid Y^1 = y^1)$ by considering probabilities $P_{X_i \to Y^1}$, $i = 4, 5, 8$ (blue arrow). (b) Message sent from a variable node to a check node. The message sent from a variable node X_2 to a check node Y^1 (red arrow) is expressed as the probability $P_{X_2 \to Y^1}(X_2 \text{ is abnormal})$ by considering probability $P_{Y^2 \to X_2}$.

graph for a given sensing matrix A in Fig. 4.1(b). In our message passing algorithm, messages are exchanged between variable nodes and check nodes.

The messages exchanged between variable nodes and check nodes are the probabilities that a random variable X_i is abnormal. More precisely, the message sent from a variable node X_i to a check node Y^j is the probability that the variable node X_i is abnormal, namely $P_{X_i \to Y^j}(X_i \text{ is abnormal})$, based on local information gathered at

variable node X_i. Similarly, the message sent from a check node Y^j to a variable node X_i is the probability that the variable node X_i is abnormal based on local information gathered at variable node Y^j. For example, Fig. 4.2(a) shows the message sent from a check node Y^1 to a variable node X_2, which is the probability that X_2 is abnormal given observation y^1 and incoming messages from X_4 to Y^1, X_5 to Y^1, and X_8 to Y^1. Fig. 4.2(b) illustrates the message from a variable node X_2 to a check node Y_1, which is the probability that X_2 is abnormal when we consider the incoming message from Y^2 to X_2. The message m from a check node Y^j to a variable node X_i is expressed as a function of incoming messages from neighbors of X_i and Y^j, and given by

$$m_{Y^j \to X_i} = f(m_{X_{l \in N(Y^j) \setminus X_i} \to Y^j}, Y^j) = P_{Y_j \to X_i}(X_i \text{ is abnormal} \mid Y_j = y_j), \tag{4.20}$$

where $f(\cdot)$ is a function calculating the probability of X_i being abnormal based on incoming messages from its neighbor variable nodes and realized observation y^j. The message m from a variable node to a check node is expressed by

$$m_{X_i \to Y^j} = h(m_{Y^{l \in N(X_j) \setminus Y^j} \to X_i}) = P_{X_i \to Y^j}(X_i \text{ is abnormal}), \tag{4.21}$$

where $h(\cdot)$ is a function calculating the probability of X_i being abnormal based on incoming messages from check nodes. In the same way, we calculate the probability that X_i is normal.

4.7 Numerical Experiments

In this section, we numerically evaluate the performance of mixed observations in hypothesis testing. We first simulate the error probability of identifying anomalous

random variables through linear mixed observations. The linear mixing used in the simulation is based on sparse bipartite graphs. In sparse bipartite graphs [12,160,189], n variable nodes on the left are used to represent the n random variables, and m measurement nodes on the right are used to represent the m measurements. If and only if the i-th random variable is nontrivially involved in the j-th measurement, there is an edge connecting the i-th variable node to the j-th measurement node. Unlike sparse bipartite graphs already used in LDPC codes, and compressed sensing [12,160,189], a novelty in this paper is that our sparse bipartite graphs are allowed to have more measurement nodes than variable nodes, namely $m \geq n$. In this simulation, there are 6 edges emanating from each measurement node on the right, and there are $\frac{6m}{n}$ edges emanating from each variable node on the left. After a uniformly random permutation, the $6m$ edges emanating from the measurement nodes are plugged into the $6m$ edge "sockets" of the left variable nodes. If there is an edge connecting the i-th variable node to the j-th measurement node, then the linear mixing coefficient before the i-th random variable in the j-th measurement is set to 1; otherwise that linear mixing coefficient is set to 0. We call the Likelihood Ratio Test (LRT) with separate observations and mixed (compressed) observations as SLRT and CLRT respectively. We also abbreviate the message passing and LASSO based hypothesis testing method to MP and LASSO in the figures below.

4.7.1 Random variables with different variances

In the first simulation example, we take $n = 100$, and let m vary from 50 to 450. We assume that $k = 1$ random variable follows the Gaussian distribution $\mathcal{N}(0, 100)$, and the other $(n - k) = 99$ random variables follow another distribution $\mathcal{N}(0, 1)$. We used likelihood ratio test algorithms to find the anomalous random variables through the described linear mixed observations based on sparse bipartite graphs. For comparison, we also implement the likelihood ratio test algorithms for separate observations of random variables, where we first make $\lfloor \frac{m}{n} \rfloor$ separate observations of each random variables, and then made an additional separate observation for uniformly randomly selected $(m \mod n)$ random variables. For each m, we perform 1000 random trials, and record the number of trials failing to identify the anomalous random variables. The error probability, as a function of m, is plotted in Fig. 4.3. We can see that linear mixed observation offers significant reduction in the error probability of hypothesis testing, under the same number of observations.

We further carry out simulations for $n = 200$. Fig. 4.4 and 4.5 show the hypothesis testing error probability when normal and abnormal random variables follow the Gaussian distributions $\mathcal{N}(0, 1)$ and $\mathcal{N}(0, 100)$ on $k = 1$ and 2 respectively. The hypothesis testing error probability is obtained from 1000 random trials. In this simulation, we use MP based hypothesis testing algorithm and compare it against LRT methods.

For large values of n and k, Fig. 4.6 shows the performance of MP based hypothesis testing algorithm when the two types of random variables have different

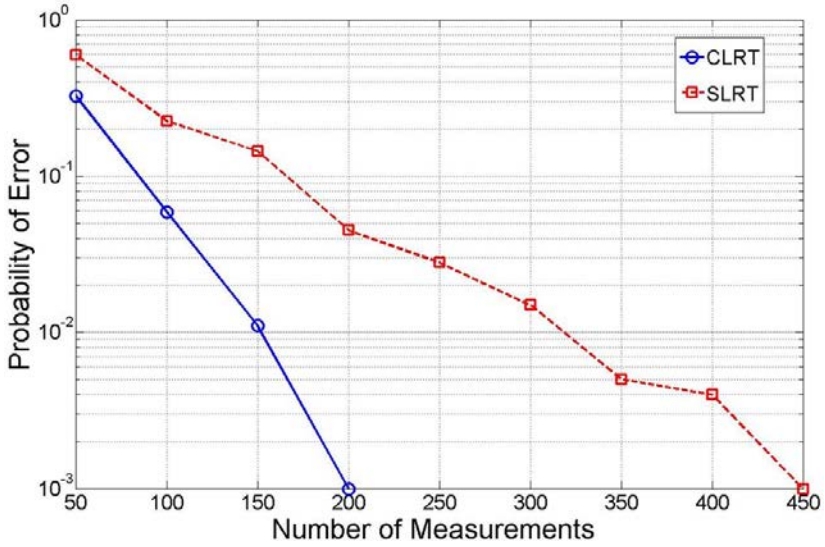


Figure 4.3: Error probability as a function of m in log scale when $(n, k) = (100, 1)$. The normal and abnormal random variables follow $\mathcal{N}(0, 1)$ and $\mathcal{N}(0, 100)$ respectively.

variances. We set k from 1 to 11 when $n = 1000$. In this parameter setup, LRT methods have difficulties in finding the k abnormal random variables out of n. The normal and abnormal random variables follow the Gaussian distribution $\mathcal{N}(0, 1)$ and $\mathcal{N}(0, 100)$ respectively, and the error probability is obtained from 500 random trials.

4.7.2 Random variables with different means

Under the same simulation setup as in Fig. 4.3, we test the error probability performance of mixed observations for two Gaussian distributions: the anomalous Gaussian distribution $\mathcal{N}(0, 1)$, and the normal Gaussian distribution $\mathcal{N}(8, 1)$. We also slightly adjust the number of total random variables as $n = 102$, to make sure that each random variable participates in the same integer number $\frac{6m}{n}$ of measurements.

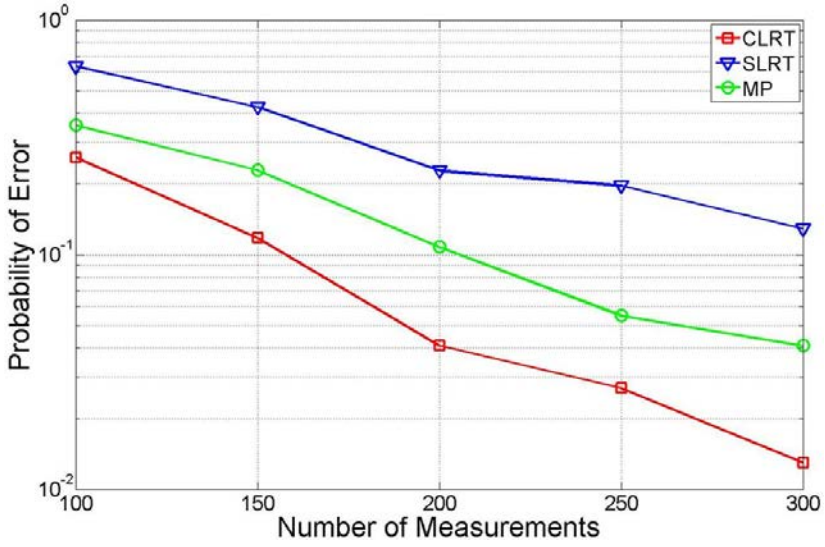


Figure 4.4: Error probability as a function of m in log scale when $(n, k) = (200, 1)$. The regular and abnormal random variables follow $\mathcal{N}(0, 1)$ and $\mathcal{N}(0, 100)$ respectively.

Mixed observations visibly reduce the error probability under the same number of measurements, compared with separate observations. For example, even when $m = 68 < n = 102$, CLRT correctly identifies the anomalous random variable in 999 out of 1000 cases by using m mixed observations from the bipartite graphs. Fig. 4.7 shows the result when the two types of random variables have different expectations.

In addition, we carry out simulations to show the results from MP and LASSO based hypothesis testing methods and compare them against the results from LRT methods. Fig. 4.8 and 4.9 have the simulation results when the normal and abnormal random variables follow the Gaussian distribution $\mathcal{N}(0, 1)$ and $\mathcal{N}(7, 1)$ on $k = 1$ and 2 respectively. The error probability is obtained from 1000 random trials.

We compare the performance of MP and LASSO based hypothesis testing algorithms on large values of n and k which is the computational challenging case for LRT

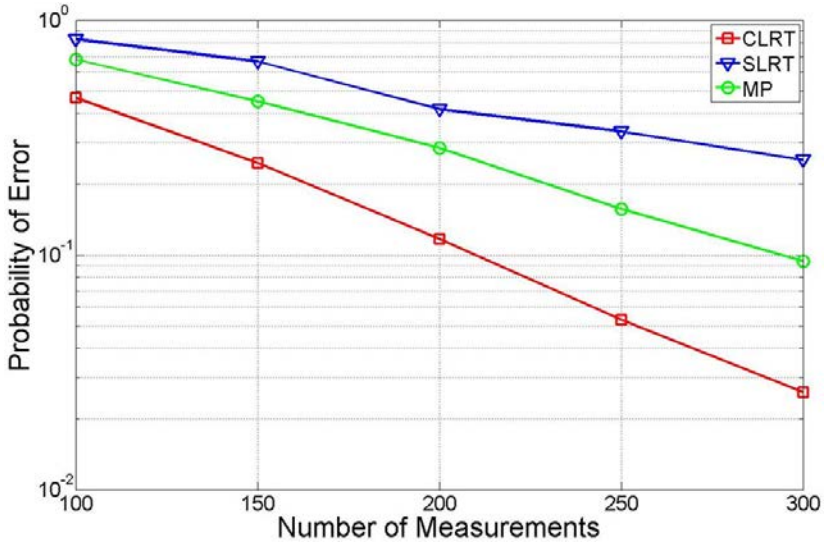


Figure 4.5: Error probability as a function of m in log scale when $(n, k) = (200, 2)$. The regular and abnormal random variables follow $\mathcal{N}(0, 1)$ and $\mathcal{N}(0, 100)$ respectively.

in Fig. 4.10 and 4.11. For this simulation, we set n to 1000 and k from 1 to 11, and for each m, we perform 500 random trials to obtain the error probability. The normal and abnormal random variables follow $\mathcal{N}(0, 1)$ and $\mathcal{N}(8, 1)$ respectively. Finally, Fig. 4.12 shows the comparison result between MP and LASSO based hypothesis testing methods.

4.7.3 Random variables with different means and variances

We further test the error probability performance of mixed observations for two Gaussian distributions with different means and variances. Fig. 4.13 and 4.14 show the results when two types of random variables have different means and variances. The abnormal and normal random variables follow $\mathcal{N}(0, 1)$, and $\mathcal{N}(7, 100)$ respectively. We obtain the error probability from 1000 random trials. The simulation results show that CLRT and MP still identify the abnormal random variables

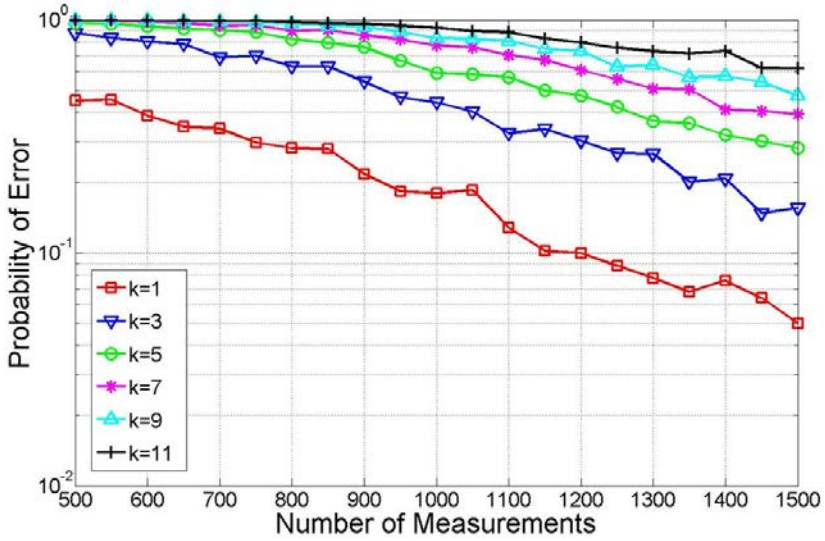


Figure 4.6: Error probability of MP based hypothesis testing algorithm as a function of m in log scale on various k values when n = 1000. The normal and abnormal random variables follow $\mathcal{N}(0,1)$ and $\mathcal{N}(0,100)$ respectively.

with fewer observations than SLRT.

4.8 Conclusion

In this paper, we have studied the problem finding k anomalous random variables following a different probability distribution among n random variables, by using *non-adaptive* mixed observations of these n random variables. Our analysis has shown that mixed observations, compared with separate observations of individual random variables, can significantly reduce the number of samples required to identify the anomalous random variables accurately. Compared with conventional compressed sensing problems, in our setting, each random variable may take dramatically different realizations in different observations.

There are some questions that remain open in performing hypothesis testing

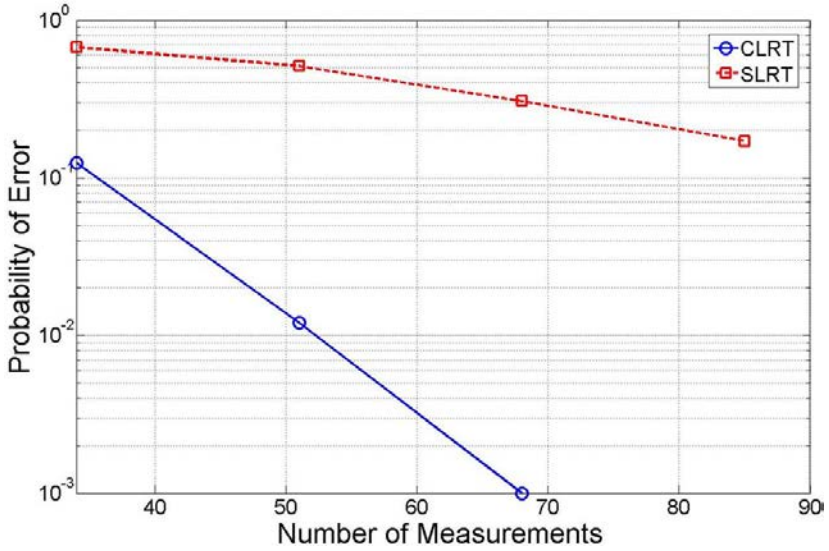


Figure 4.7: Error probability as a function of m in log scale when n = 102. The normal and abnormal random variables follow $\mathcal{N}(8,1)$ and $\mathcal{N}(0,1)$ respectively.

from mixed observations. For example, for random variables of non-Gaussian distributions, it is not explicitly known what linear mixed observations maximize the anomaly detection error exponents. In addition, it is very interesting to explore the mixed observations for anomaly detection for random variables with unknown abnormal probability distributions.

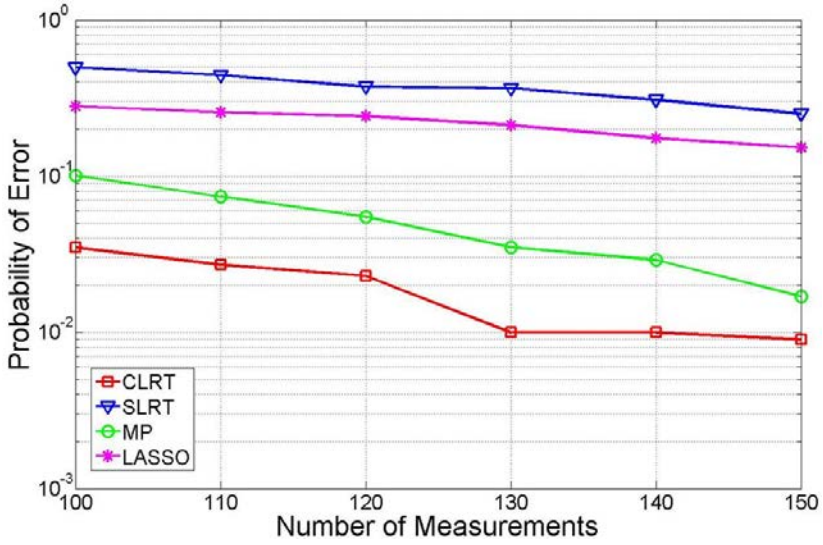


Figure 4.8: Error probability as a function of m in log scale when $(n, k) = (200, 1)$. The normal and abnormal random variables follow $\mathcal{N}(0, 1)$ and $\mathcal{N}(7, 1)$ respectively.

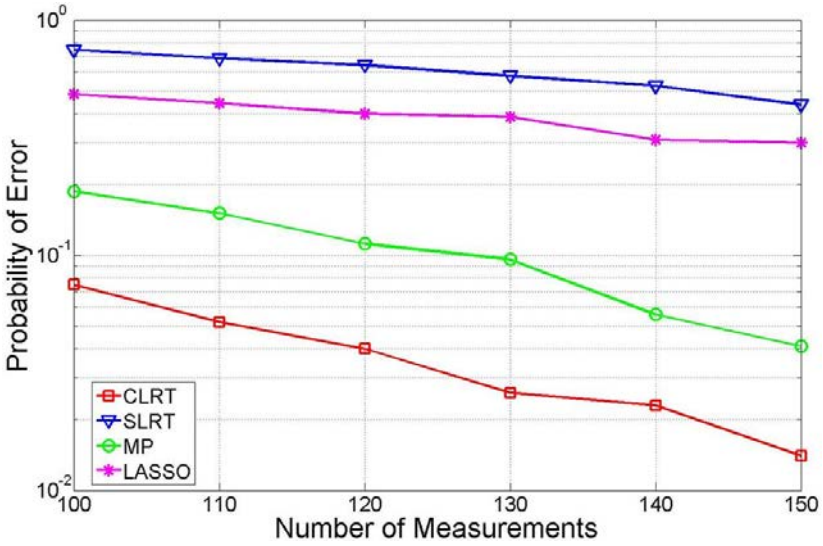


Figure 4.9: Error probability as a function of m in log scale when $(n, k) = (200, 2)$. The normal and abnormal random variables follow $\mathcal{N}(0, 1)$ and $\mathcal{N}(7, 1)$ respectively.

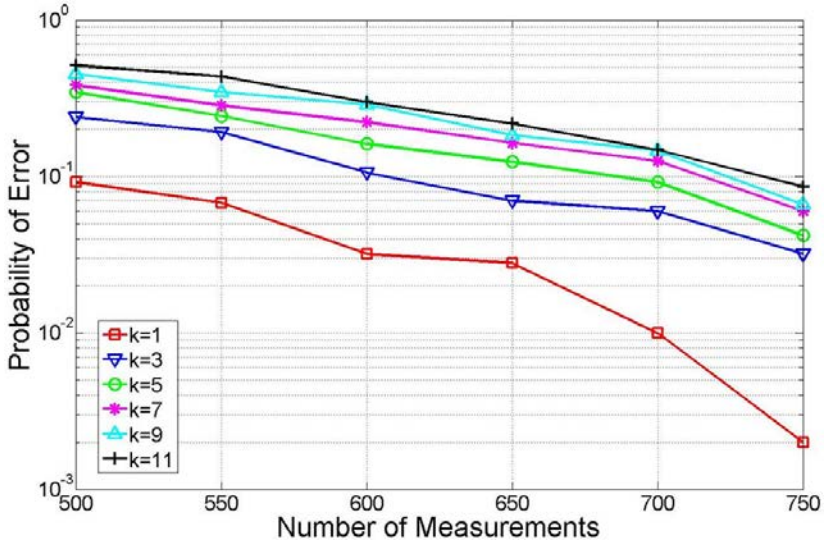


Figure 4.10: Error probability of MP based hypothesis testing algorithm as a function of m in log scale on various k values when n = 1000. The regular and abnormal random variables follow $\mathcal{N}(0,1)$ and $\mathcal{N}(8,1)$ respectively.

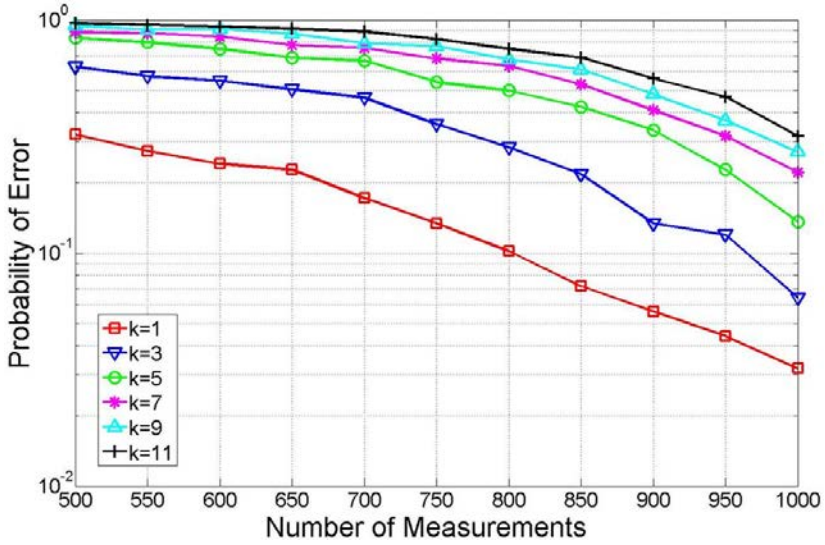


Figure 4.11: Error probability of LASSO based hypothesist testing algorithm as a function of m in log scale on various k values when n = 1000. The regular and abnormal random variables follow $\mathcal{N}(0,1)$ and $\mathcal{N}(8,1)$ respectively.

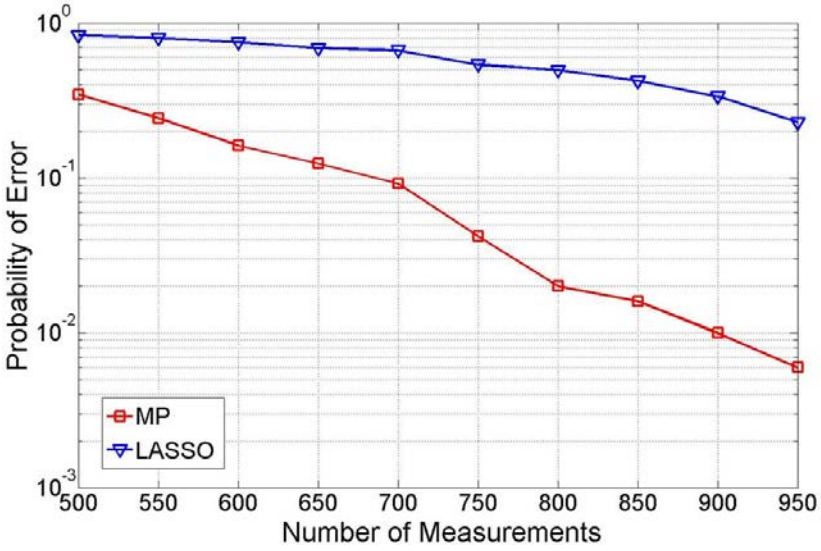


Figure 4.12: Error probability as a function of m in log scale on various k values when $(n, k) = (1000, 5)$. The regular and abnormal random variables follow $\mathcal{N}(0, 1)$ and $\mathcal{N}(8, 1)$ respectively.

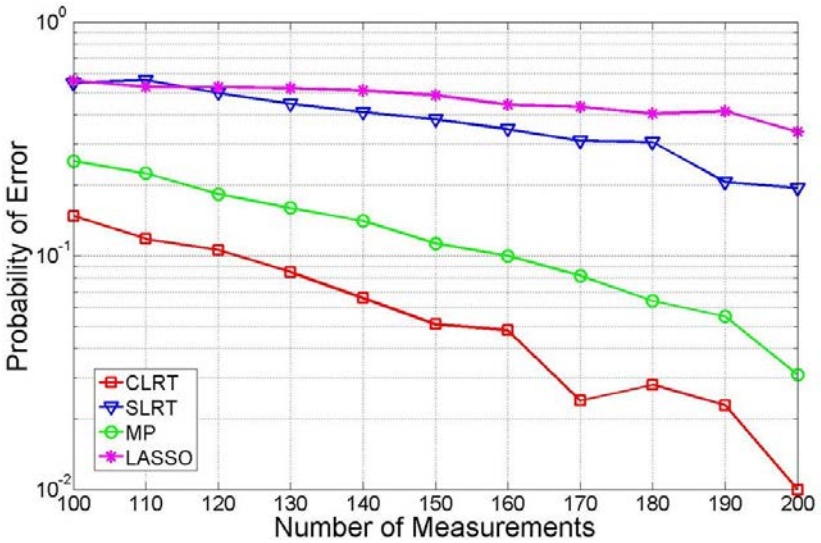


Figure 4.13: Error probability as a function of m in log scale when $(n, k) = (200, 1)$. The regular and abnormal random variables follow $\mathcal{N}(0, 1)$ and $\mathcal{N}(7, 100)$ respectively.

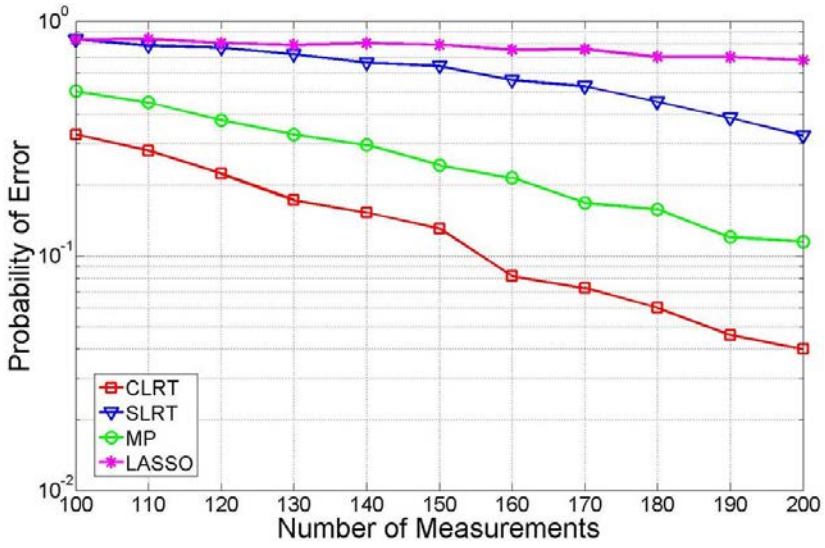


Figure 4.14: Error probability as a function of m in log scale when $(n, k) = (200, 2)$. The regular and abnormal random variables follow $\mathcal{N}(0, 1)$ and $\mathcal{N}(7, 100)$ respectively.

CHAPTER 5
NETWORK CONSTRAINED DISTRIBUTED DUAL COORDINATE ASCENT FOR MACHINE LEARNING

5.1 Introduction

In modern society, the abundance of sensors enables data to be ubiquitously collected at an astonishing speed. The amount of data that we can access and learn actionable information from is skyrocketing. This propels our society into an era of *big data*. "Big data" is dramatically impacting our everyday lives, and almost every aspect of our society, including education, science, engineering, finance, healthcare, and management [42]. The Google flu trend service is one example of using big data for better healthcare services [89].

To make use of big data, one often uses large-scale convex optimizations to learn actionable information from these big data [106]. However, very often big data are collected and stored at different locations. Due to the constraints of limited storage volumes and network communication bandwidths, sometimes it is not possible to pool these distributed data at one central location, and then perform centralized machine learning over these data. This necessitates performing machine learning in a distributed manner.

In order to overcome this obstacle of distributed data, researchers have studied various optimization methods such as synchronous Stochastic Gradient Decent (SGD) [41], synchronous Stochastic Dual Coordinate Ascent (SDCA) [106,192], asynchronous SGD [200], and asynchronous SDCA [97,102]. Even though the convergence

of SGD does not depend on the size of data, it is reported in [96] that SDCA can outperform SGD when we need relatively high solution accuracy. Furthermore, asynchronous updating scheme can suffer from the conflicts between intermediate results.

Motivated by these facts, the authors in [106, 192] considered a synchronous distributed dual coordinate ascent algorithm for solving regularized loss minimization problems, in a star network. In this star network, data are distributed over a few local workers, which can individually communicate with the central station. In [106, 192], the authors derived the convergence rate of the distributed dual coordinate ascent algorithm, with respect to the number of iterations. The proposed distributed optimization framework in [106] is free of tuning parameters or learning rates, compared with SGD-based methods. Moreover, the duality gap in [106] readily provides a fair stopping criterion and efficient accuracy certificates. In [97, 102], the authors investigated the performance of asynchronous SDCA.

In practical communication networks, computers are not always organized in a star work, and can have very different network topologies. It is unclear how to design and analyze dual coordinate ascent algorithms for a network with general topologies. In addition, it is unknown how network communication delays (not merely the number of communication rounds) will impact the design and convergence rate of distributed dual coordinate ascent algorithms [106, 192]. We remark that, in [176], the authors considered communication delays and provided the convergence bound in terms of time for consensus based distributed optimization.

In this paper, we study how the network communication constraints will im-

pact the convergence speed of distributed dual coordinate ascent optimization algorithms. Firstly, we consider the design and analysis of distributed dual coordinate ascent algorithms for regularized loss minimization, in a general *tree structured* network. We give the convergence rate analysis of the distributed dual coordinate ascent algorithm for the considered tree network. Secondly, by considering network communication delays, we optimize the network-constrained dual coordinate ascent to maximize its convergence speed. Our results show that under different network communication delays, to achieve maximum convergence speed, one needs to adopt delay-dependent number of local iterations for distributed dual coordinate ascent algorithms.

Notations: We denote the set of real numbers as $\mathbb{R}$. We use $[k]$ to denote the index set of the coordinates in the k-th coordinate block. For a index set Q, we use $\overline{Q}$ and $|Q|$ to represent the complement and the cardinality of Q respectively. We use bold letters to represent vectors. If we use an index set as a subscript of a vector (matrix), we refer to the partial vector (partial matrix) over the index set (with columns over the index set). The superscript (t) is used to denote the t-th iteration. For example, $\boldsymbol{\alpha}_{[k]}^{(t)}$ represents a partial vector $\boldsymbol{\alpha}$ over the k-th block coordinate at the t-th iteration. We use the superscript $\star$ to denote the optimal solution.

5.2 Problem Setup

We have the following regularized loss minimization problem for machine learning applications [97, 102, 106, 159, 192]:

$$\underset{\boldsymbol{w}\in\mathbb{R}^d}{\text{minimize}}\ P(\boldsymbol{w}) \triangleq \frac{\lambda}{2}\|\boldsymbol{w}\|^2 + \frac{1}{m}\sum_{i=1}^{m}\ell_i(\boldsymbol{w}^T\boldsymbol{x}_i), \tag{5.1}$$

where $\boldsymbol{x}_i \in \mathbb{R}^d$, $i = 1, ..., m$, are dataset, $\ell_i(\cdot)$, $i = 1, ..., m$, are loss functions, and λ is the regularization parameter. Depending on the loss functions, one can consider (5.1) as various machine learning problems ranging from regression to classification. For example, if the loss function is the hinge loss function, the optimization problem with labeled dataset $\{(\boldsymbol{x}_i, y_i)\}$, $i = 1, ..., m$, where $y_i \in \mathbb{R}$ is label information, is the Support Vector Machine (SVM).

By using the conjugate function, i.e., $\ell_i(a) = \sup_b ab - \ell_i^*(b)$, where $a, b \in \mathbb{R}$ and $\ell_i(\cdot)$ is convex, we can obtain the dual problem of (5.1) as follows:

$$\underset{\boldsymbol{\alpha}\in\mathbb{R}^m}{\text{maximize}}\ D(\boldsymbol{\alpha}) \triangleq -\frac{\lambda}{2}\|A\boldsymbol{\alpha}\|^2 - \frac{1}{m}\sum_{i=1}^{m}\ell_i^*(-\alpha_i), \tag{5.2}$$

where α_i is the i-th element of the dual vector $\boldsymbol{\alpha}$, and the data matrix $A \in \mathbb{R}^{d\times m}$ has the normalized training data $\frac{1}{\lambda m}\boldsymbol{x}_i$ in its i-th column; namely $A_i = \frac{1}{\lambda m}\boldsymbol{x}_i$. Since we have the primal-dual relationship as $\boldsymbol{w}(\boldsymbol{\alpha}) \triangleq A\boldsymbol{\alpha}$, we have the duality gap as $P(\boldsymbol{w}(\boldsymbol{\alpha})) - D(\boldsymbol{\alpha})$.

In this paper, we consider data being distributed over a network of computers.

5.3 Review of Distributed Dual Coordinate Ascent for a Star Network

The authors in [102, 106, 192] consider a star network as shown in Fig. 5.1 and assume that each local worker has disjoint parts of dataset. Specifically, the k-th local worker has training data $\{(\boldsymbol{x}_i, y_i)\}$, $i \in [k]$, where $[k]$ is the index set for the training data of the k-th local worker. Hence, if the star network has K local workers, $|\cup_{k=1}^{K}[k]| = m$. In [106], the author introduced Algorithm 5.1 for the

Algorithm 5.1 Distributed Dual Coordinate Ascent [106]

Input: $T \geq 1$
Output: $\boldsymbol{w}$, $\boldsymbol{\alpha}$
Data: $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^m$ distributed over K local workers
Initialization: $\boldsymbol{\alpha}_{[k]}^{(0)} \leftarrow 0$ for all local workers, and $\boldsymbol{w}^{(0)} \leftarrow 0$
for $t = 1$ **to** T **do**
  for *all local workers* $k = 1, 2, ..., K$ *in parallel* **do**
    $(\triangle \boldsymbol{\alpha}_{[k]}, \triangle \boldsymbol{w}_k) \leftarrow \text{LocalDualMethod}(\boldsymbol{\alpha}_{[k]}^{(t-1)}, \boldsymbol{w}^{(t-1)})$
    $\boldsymbol{\alpha}_{[k]}^{(t)} \leftarrow \boldsymbol{\alpha}_{[k]}^{(t-1)} + \frac{1}{K} \triangle \boldsymbol{\alpha}_{[k]}$
  end
  send $\triangle \boldsymbol{w}_k$, $k = 1, ..., K$, to the central station
  $\boldsymbol{w}^{(t)} \leftarrow \boldsymbol{w}^{(t-1)} + \frac{1}{K} \sum_{k=1}^K \triangle \boldsymbol{w}_k$
  distribute $\boldsymbol{w}^{(t)}$ to local workers
end

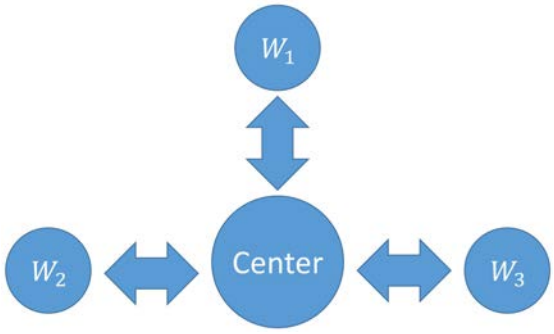


Figure 5.1: Illustration of a distributed star network, where W_i, $i = 1, 2, 3$, are local workers.

distributed dual coordinate ascent. In Algorithm 5.1, LocalDualMethod(·) represents any dual method to solve (5.2). The Stochastic Dual Coordinate Ascent (SDCA), denoted by LocalSDCA(·), is a possible candidate for LocalDualMethod(·) [106]. The convergence rate of the algorithm is given as follows [106].

Theorem 5.1 ([106, Theorem 2])**.** *Assume that Algorithm 5.1 is run for T outer iterations of K local computers, with the procedure LocalSDCA(·) having local geomet-*

ric improvement Θ. *Further, assume the loss functions* $\ell_i(\cdot)$ *are* $1/\gamma$*-smooth. Then, the following geometric convergence rate holds for the global (dual) objective:*

$$E[D(\boldsymbol{\alpha}^*) - D(\boldsymbol{\alpha}^{(T)})] \leq \left(1 - (1-\Theta)\frac{1}{K}\frac{\lambda m \gamma}{\rho + \lambda m \gamma}\right)^T \left(D(\boldsymbol{\alpha}^*) - D(\boldsymbol{\alpha}^{(0)})\right), \tag{5.3}$$

where ρ *is any real number satisfying*

$$\rho \geq \rho_{min} \triangleq \underset{\boldsymbol{\alpha} \in \mathbb{R}^m}{maximize} \ \lambda^2 m^2 \frac{\sum_{k=1}^K \|A_{[k]}\boldsymbol{\alpha}_{[k]}\|^2 - \|A\boldsymbol{\alpha}\|^2}{\|\boldsymbol{\alpha}\|^2} \geq 0.$$

For LocalSDCA$(\cdot)$, Θ can be set to the following value with $s \in [0,1]$ [106]:

$$\Theta = \left(1 - \frac{s}{\tilde{m}}\right)^H, \tag{5.4}$$

where $\tilde{m} \triangleq \max_{k=1,\ldots,K} m_k$ is the size of the largest block of coordinates among K local workers, H is the number of local (or inner) iterations in LocalSDCA$(\cdot)$, and $s \in [0,1]$ is a step size which determines how far the next solution will be from the current solution at each iteration.

5.4 Generalized Distributed Coordinate Ascent Algorithms for Tree-structured Networks

Earlier works [106, 192] provide the convergence analysis for distributed dual coordinate ascent algorithms in a star network as illustrated in Fig. 5.1. However, the communication network connecting different local workers is not necessarily a simple star network, but instead can be an arbitrary undirected connected graph. The design and analysis of distributed dual coordinate ascent algorithm in a communication network of general topology is not well understood [192]. One may argue that, in a connected communication network, we can always form a virtual star network by

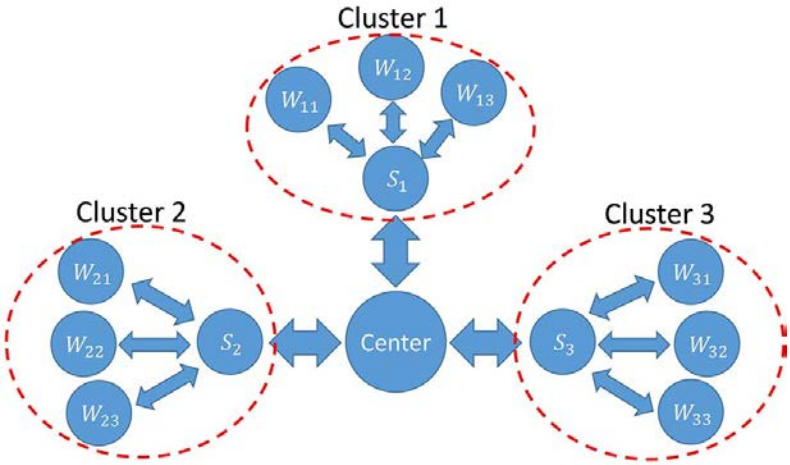


Figure 5.2: A tree-structured network, which has two layers. In the network, the center (root node) has 3 direct child nodes S_1, S_2 and S_3. Each node S_i has 3 direct local workers W_{ij}, $j = 1, 2, 3$.

connecting local workers to the central station through the relays of other computers. However, the communication delay from one particular local worker to the central station can be very large (long relays), significantly slowing down the convergence of the distributed learning algorithm. Thus it is necessary to spend more computational resources on performing distributed optimization among local workers close to each other first, before communicating intermediate computational results to a central station.

Motivated by these network constraints, in this section, we investigate the design and analysis of a recursive distributed dual coordinate ascent algorithm over a general tree structured network, instead of a star network. We choose to investigate the tree network, because every connected communication network has a spanning tree. In addition, the tree structured network is a generalization of the star network.

We first describe a general tree network, with a 2-layer tree network example illustrated in Fig. 5.2. In the considered tree network, the root node corresponds to the central station. Any other tree node corresponds to a local worker. Each tree node may have several direct child nodes. Without loss of generality, we assume that only the local workers corresponding to the leaf nodes have access to the distributed data, namely disjoint segmented blocks of the data matrix A (In fact, if a non-leaf node Q stores data, one can create a virtual leaf node L attached to Q, and "stores" data in L). For a tree node Q, we also use Q to denote the set of indices of data points stored in the subtree with Q as the root node (the subtree includes Q, indirect and direct child nodes of Q). We use $[Q, k]$ to denote the set of indices of data points stored in the subtree whose root node is the k-th direct child node of Q. If Q is a leaf node, we use m_Q to denote the number of data points stored in Q. In a tree network, a node can only communicate with its child nodes or parent nodes.

We are ready to introduce the generalized distributed dual coordinate ascent algorithm (which we call TreeDualMethod) for solving (5.2) dealing with data stored in a general tree structure network. Algorithm 5.2, Algorithm 5.3 and Procedure A describe respectively the computational steps of TreeDualMethod for a general tree node (not root or leaf), the root node, and a leaf node.

5.5 Convergence Analysis of TreeDualMethod for a Tree Network

In this section, we show that for a tree network, there is a recursive relation between the convergence rate of the algorithm at a tree node Q, and the convergence

Algorithm 5.2 TreeDualMethod: General Distributed Dual Coordinate Ascent for a General Tree Node Q (not root or leaf)

Input: $T \geq 1$, $\boldsymbol{\alpha}_Q$, $\boldsymbol{w}$

Initialization: $\boldsymbol{\alpha}_{[Q,k]}^{(0)} \leftarrow \boldsymbol{\alpha}_{[Q,k]}$ for all direct child nodes k of node Q , $\boldsymbol{w}^{(0)} \leftarrow \boldsymbol{w}_Q$

for $t = 1$ **to** T **do**

  for *all direct child nodes* $k = 1, 2, ..., K$ *of* Q *in parallel* **do**

    $(\triangle \boldsymbol{\alpha}_{[Q,k]}, \triangle \boldsymbol{w}_k) \leftarrow \text{TreeDualMethod}(\boldsymbol{\alpha}_{[Q,k]}^{(t-1)}, \boldsymbol{w}^{(t-1)})$

    $\boldsymbol{\alpha}_{[Q,k]}^{(t)} \leftarrow \boldsymbol{\alpha}_{[Q,k]}^{(t-1)} + \frac{1}{K} \triangle \boldsymbol{\alpha}_{[Q,k]}$

  end

  reduce $\boldsymbol{w}^{(t)} \leftarrow \boldsymbol{w}^{(t-1)} + \frac{1}{K} \sum_{k=1}^{K} \triangle \boldsymbol{w}_k$

end

Output: $\triangle \boldsymbol{\alpha}_Q \triangleq \boldsymbol{\alpha}_Q^{(T)} - \boldsymbol{\alpha}_Q^{(0)}$, and $\triangle \boldsymbol{w}_Q \triangleq \boldsymbol{w}^{(T)} - \boldsymbol{w}^{(0)} = A_Q \triangle \boldsymbol{\alpha}_Q$

Algorithm 5.3 TreeDualMethod: General Distributed Dual Coordinate Ascent for the Root Node Q

Input: $R \geq 1$

Initialization: $\boldsymbol{\alpha}_{[Q,k]}^{(0)} \leftarrow 0$ for all direct child nodes k of node Q, $\boldsymbol{w}^{(0)} \leftarrow 0$

for $t = 1$ **to** R **do**

  for *all direct child nodes* $k = 1, 2, ..., K$ *in parallel* **do**

    $(\triangle \boldsymbol{\alpha}_{[Q,k]}, \triangle \boldsymbol{w}_k) \leftarrow \text{TreeDualMethod}(\boldsymbol{\alpha}_{[Q,k]}^{(t-1)}, \boldsymbol{w}^{(t-1)})$

    $\boldsymbol{\alpha}_{[Q,k]}^{(t)} \leftarrow \boldsymbol{\alpha}_{[Q,k]}^{(t-1)} + \frac{1}{K} \triangle \boldsymbol{\alpha}_{[Q,k]}$

  end

  reduce $\boldsymbol{w}^{(t)} \leftarrow \boldsymbol{w}^{(t-1)} + \frac{1}{K} \sum_{k=1}^{K} \triangle \boldsymbol{w}_k$

end

Output: $\boldsymbol{\alpha}^{(R)}$, and $\boldsymbol{w}^{(R)}$

rate of the algorithm at Q's direct child nodes.

We assume that Q has K direct child nodes, and denote the dual variable vector corresponding to its k-th direct child node by $\boldsymbol{\alpha}_{[Q,k]}$, $1 \leq k \leq K$. We define the local suboptimality gap for Q's k-th direct child node as:

Procedure A. TreeDualMethod: General Distributed Dual Coordinate Ascent for a Leaf Tree Node Q

Input: $H \geq 1$, $\boldsymbol{\alpha}_Q \in \mathbb{R}^{m_Q}$, and $\boldsymbol{w} \in \mathbb{R}^d$ consistent with other coordinate blocks of $\boldsymbol{\alpha}$ s.t. $\boldsymbol{w} = A\boldsymbol{\alpha}$
Data: $\{(x_i, y_i)\}_{i=1}^{m_Q}$
Initialization: $\triangle\boldsymbol{\alpha}_Q \leftarrow 0 \in \mathbb{R}^{m_Q}$, and $\boldsymbol{w}^{(0)} \leftarrow \boldsymbol{w}$
for $h = 1$ **to** H **do**
  choose $i \in \{1, 2, ..., m_Q\}$ uniformly at random
  find $\triangle\alpha$ maximizing $-\frac{\lambda m}{2}\|\boldsymbol{w}^{(h-1)} + \frac{1}{\lambda m} \triangle \alpha \boldsymbol{x}_i\|^2 - \ell_i^*(-(\alpha_i^{(h-1)} + \triangle\alpha))$
  $\alpha_i^{(h)} \leftarrow \alpha_i^{(h-1)} + \triangle\alpha$
  $(\triangle\boldsymbol{\alpha}_Q)_i \leftarrow (\triangle\boldsymbol{\alpha}_Q)_i + \triangle\alpha$
  $\boldsymbol{w}^{(h)} \leftarrow \boldsymbol{w}^{(h-1)} + \frac{1}{\lambda m} \triangle \alpha \boldsymbol{x}_i$
end
Output: $\triangle\boldsymbol{\alpha}_Q$ and $\triangle\boldsymbol{w}_k \triangleq A_Q \triangle \boldsymbol{\alpha}_Q$

$$\epsilon_{Q,k}(\boldsymbol{\alpha}) \triangleq \underset{\hat{\boldsymbol{\alpha}}_{[Q,k]}}{\text{maximize}}\, D((\boldsymbol{\alpha}_{[Q,1]}, ..., \hat{\boldsymbol{\alpha}}_{[Q,k]}, ..., \boldsymbol{\alpha}_{[Q,K]}, \boldsymbol{\alpha}_{\overline{Q}})) - D((\boldsymbol{\alpha}_{[Q,1]}, ..., \boldsymbol{\alpha}_{[Q,k]}, ..., \boldsymbol{\alpha}_{[Q,K]}, \boldsymbol{\alpha}_{\overline{Q}})), \tag{5.5}$$

Note that the suboptimality gap for the k-th child node is defined when $\boldsymbol{\alpha}_{[Q,i]}$'s $(i \neq k)$ and $\alpha_{\overline{Q}}$ are fixed. We further assume that we have the following local geometric improvement for the k-th direct child node of Q.

Assumption 5.5.1 (Direct child node geometric improvement of TreeDualMethod)**.** *Let us consider a tree node Q. We assume that there exists $\Theta \in [0, 1)$ such that for any given $\boldsymbol{\alpha}$, TreeDualMethod for Q's k-th direct child node returns an update $\triangle\boldsymbol{\alpha}_{[Q,k]}$ such that*

$$E[\epsilon_{Q,k}((\boldsymbol{\alpha}_{[Q,1]}, ..., \boldsymbol{\alpha}_{[Q,k-1]}, \boldsymbol{\alpha}_{[Q,k]} + \triangle\boldsymbol{\alpha}_{[Q,k]}, ..., \boldsymbol{\alpha}_{[Q,K]}, \boldsymbol{\alpha}_{\overline{Q}}))] \leq \Theta \cdot \epsilon_{Q,k}(\boldsymbol{\alpha}). \tag{5.6}$$

We remark that this geometric improvement condition holds true if the k-th direct child node of Q is a leaf child node. For a leaf node, TreeDualMethod uses

LocalSDCA in [106] as described in Procedure A. The following proposition [106] gives a bound on the convergence for a leaf node B (even when the input $\boldsymbol{w}$ to Procedure A is determined also by $\boldsymbol{\alpha}_{\overline{Q}}$ and $\boldsymbol{\alpha}_{Q \setminus B}$).

Proposition 5.2 ([106])**.** *Let us consider a tree node Q whose direct child node B is a leaf node. Assume loss functions $\ell_i(\cdot)$ are $1/\gamma$-smooth. Then for leaf node B, Assumption 5.5.1 holds with*

$$\Theta = \left(1 - \frac{\lambda m \gamma}{1 + \lambda m \gamma} \frac{1}{m_B}\right)^H. \tag{5.7}$$

where m_B is the size of data stored at node B.

Our new result Theorem 5.3 shows that if the geometric improvement condition holds true for direct child nodes of Q, then the geometric improvement condition also holds true for Q, thus leading to a recursive calculation of the convergence rate for the tree network.

Theorem 5.3. *Let us consider a tree node Q which has K direct child nodes. The K direct child nodes satisfy the local geometric improvement requirement Assumption 5.5.1, with parameters Θ_1, Θ_2, ..., and Θ_K. We assume that Algorithm 5.2 (or Algorithm 5.3) has an input $\boldsymbol{w}$, and Algorithm 5.2 (or Algorithm 5.3) is run for T iterations. We further make the assumption that the loss functions $\ell_i(\cdot)$ are $1/\gamma$-smooth.*

Then, for any input $\boldsymbol{w}$ to Algorithm 5.2 (or Algorithm 5.3), the following geometric convergence rate holds for Q:

$$E[D(\boldsymbol{\alpha}_Q^*, \boldsymbol{\alpha}_{\overline{Q}}) - D(\boldsymbol{\alpha}_Q^{(T)}, \boldsymbol{\alpha}_{\overline{Q}})]$$
$$\leq \left(1 - (1-\Theta)\frac{1}{K}\frac{\lambda m\gamma}{\rho + \lambda m\gamma}\right)^T \left(D(\boldsymbol{\alpha}_Q^*, \boldsymbol{\alpha}_{\overline{Q}}) - D(\boldsymbol{\alpha}_Q^{(0)}, \boldsymbol{\alpha}_{\overline{Q}})\right), \tag{5.8}$$

where $\Theta = \max_k \Theta_k$, *and* ρ *is any real number satisfying*

$$\rho \geq \rho_{min} \triangleq \underset{\boldsymbol{\alpha} \in \mathbb{R}^{|Q|}}{maximize}\ \lambda^2 m^2 \frac{\sum_{k=1}^K \|A_{[Q,k]}\boldsymbol{\alpha}_{[Q,k]}\|^2 - \|A_Q\boldsymbol{\alpha}_Q\|^2}{\|\boldsymbol{\alpha}_Q\|^2} \geq 0.$$

Because Theorem 5.3 works for any non-leaf tree node, by combining it with Proposition 5.2, we can recursively obtain the convergence rate of the generalized distributed dual coordinate ascent algorithm for the whole tree network (noting that $\left(1 - (1-\Theta)\frac{1}{K}\frac{\lambda m\gamma}{\rho+\lambda m\gamma}\right)^T$ is the "Θ" for Q, seen by the direct parent node of Q).

Theorem 5.3 is different from Theorem 2 of [106] in two aspects. Firstly, Theorem 5.3 works for any tree node in a general tree network, beyond the star network discussed in [106]. Secondly, Theorem 5.3 is true, even when the input $\boldsymbol{w}$ of Algorithm 5.2 is not only determined by $\boldsymbol{\alpha}_Q$, but also determined by $\boldsymbol{\alpha}_{\overline{Q}}$. To see this, we note that, at the root node, $\boldsymbol{w} = A_Q\boldsymbol{\alpha}_Q + A_{\overline{Q}}\boldsymbol{\alpha}_{\overline{Q}}$, and the root node will pass $\boldsymbol{w}$ to tree node Q by recalling TreeDualMethod$(\cdot)$ for the root node's child nodes. Our proof of Theorem 5.3 addresses this challenge that the input $\boldsymbol{w}$ is also affected by $\boldsymbol{\alpha}_{\overline{Q}}$. Due to the space limitations, we omit the proof in this paper.

So far we have discussed how the network topology can affect the convergence speed of distributed dual coordinate ascent algorithms. In the next section, we consider how the communication delays, another major network constraint, impact the convergence of distributed dual coordinate ascent algorithms. By considering communication delays, we optimize the number of local iterations H for maximum

convergence speed.

5.6 Impacts of Communication Delays on the Convergence Rate of Distributed Dual Coordinate Ascent Algorithms

Earlier works [106, 192] bounded the convergence of distributed dual coordinate ascent algorithms with respect to the number of inner and outer iterations. However, in distributed dual coordinate ascent algorithms, there may be significant communication delays between computers. Thus the convergence of distributed dual coordinate ascent algorithms not only depends on how many iterations of these algorithm have been run, but also depend on the communication delays in performing these iterations. Thus we aim to investigate the convergence of distributed dual coordinate ascent algorithms with respect to total time used, including computational time and communication delays.

In this paper, for simplicity, we consider the star network as shown in Fig. 5.1 and the corresponding Algorithm 5.1, even though our analysis can be generalized to a tree network. We assume that the round-trip communication delay between a local worker and the central station is t_{delay}. Intuitively, If t_{delay} is close to 0, the local worker might want to perform a small number of local iterations, and communicate with the central station at a higher frequency; on the other hand, if t_{delay} is large, there is a large communication cost and, to speed up convergence, local workers may want to perform more local iterations before communicating with the central station. We use t_{lp} to denote the computational time for one local iteration at a worker, and

use t_{cp} to denote the computational time for one parameter update at the central station.

We assume each local worker performs H local iterations before communicating with the central station, and, in total, there are T outer iterations. The total experienced time is given by

$$t_{total} = (t_{lp}H + t_{delay} + t_{cp}) \cdot T. \tag{5.9}$$

Hence, the number T of outer iterations is given by

$$T = {t_{total}}/{(t_{lp}H + t_{delay} + t_{cp})}. \tag{5.10}$$

In (5.8), for T outer iterations, the expected gap between the optimal objective value and the current objective value for Algorithm 5.1 is given by :

$$\left(1 - \left(1 - \left[1 - \delta\right]^{H}\right)\frac{C}{K}\right)^{T}, \tag{5.11}$$

where $\delta = \frac{s}{\tilde{m}}$ and $C = {\lambda m\gamma}/{(\rho + \lambda m\gamma)}$.

Our goal is to minimize the gap in objective value (5.11) under a given total time t_{total}, by optimizing over the number H of local iterations. Hence, by plugging (5.10) into (5.11) , we obtain the following optimization problem

$$\underset{H}{\text{minimize}} \left(1 - \left(1 - \left[1 - \delta\right]^{H}\right)\frac{C}{K}\right)^{\frac{t_{total}}{t_{lp}H + t_{delay} + t_{cp}}}. \tag{5.12}$$

Through (5.12), we can obtain the fastest convergence speed by adjusting H according to t_{cp}, t_{delay}, and t_{lp}. Our results as shown in the numerical section show that, as t_{delay} increases, the optimal number H of local iterations also increases.

5.7 Numerical Experiments

In this section, we give numerical results showing how communication delays impact the convergence of distributed dual coordinate ascent algorithms.

Fig. 5.3 shows the optimal number H of local iterations, from optimizing (5.12). In the optimization, we set $(C, K, \delta, t_{total}, t_{lp}, t_{cp}) = (0.5, 3, 1/300, 1, 4 \times 10^{-5}, 3 \times 10^{-5})$, which were measured in the numerical experiments for the star network model (unit of time is second). We set $t_{delay} = r \times t_{lp}$, where r is a parameter indicating how severe the communication delay is. Figure 5.3 (a) shows the objective values of (5.12) when H is varied from 1 to 2000. The red line represents the optimal convergence bound at the optimal number of local iterations. Fig. 5.3 (b) shows the optimal number of local iterations to achieve the fastest convergence rate for different communication delays, where r is varied from 0 to 10e10.

We further experimented with solving synthetic linear regression problems for the star network shown in Fig. 5.1, in order to demonstrate how communication delays affect the optimal number of local iterations. We generated dataset $A \in \mathbb{R}^{100 \times 600}$ with i.i.d. zero-mean unit-variance Gaussian elements. We assigned evenly divided dataset to 3 local workers, i.e., 200 vectors for each local worker. We measured the average processing time t_{lp} for one iteration by running LocalSDCA over the 3 disjoint dataset. And t_{delay} is set as $t_{delay} = r \times t_{lp}$, with $r = 10$ and $10e5$ for Fig. 5.4 (a) and Fig. 5.4 (b) respectively. For fixed r, we varied the number H of local iterations from 10 to 10e4. When $t_{delay} = 10e5 \times t_{lp}$, we can see that the best number of local iterations H is 1000 or 10000. When $t_{delay} = 10 \times t_{lp}$, $H = 100$ or 1000 provides the best convergence

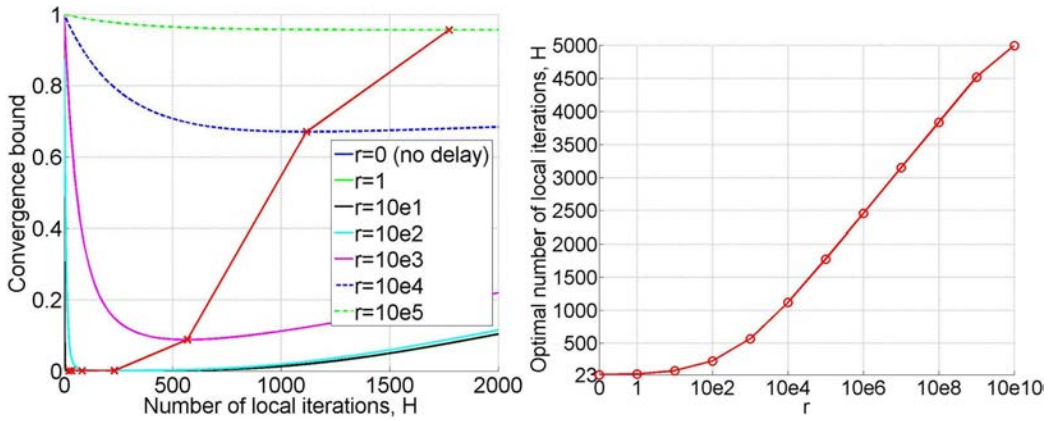
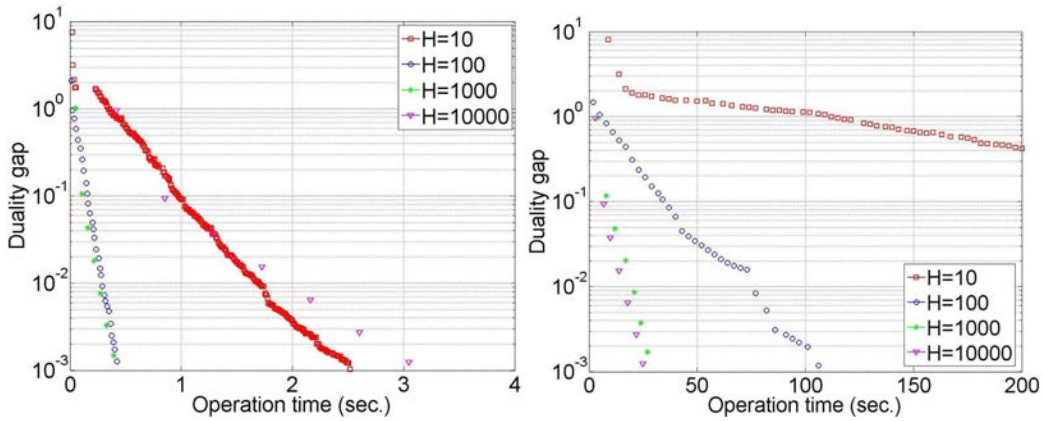


Figure 5.3: (a) The objective value of (5.12) when the number of iterations H is varied from 1 to 2000, where $(C, K, \delta, t_{total}, t_{lp}, t_{cp}) = (0.5, 3, 1/300, 1, 4 \times 10^{-5}, 3 \times 10^{-5})$ and $t_{delay} = r \times t_{lp}$. The red line represents the optimal number of local iterations to achieve the fastest convergence rate. (b) Optimal number of iterations to achieve the fastest convergence rate, when the parameters are the same as (a) and r is varied from 0 (no delay) to 10^{10}.

results. Those numerical experiments are consistent with our predictions in Fig. 5.3.



(a) $r = 10e1$ (b) $r = 10e5$

Figure 5.4: Convergence rate in terms of operation time with different $t_{delay} = r \times t_{lp}$ in communication.

CHAPTER 6
FAST DOSE OPTIMIZATION FOR ROTATING SHIELD BRACHYTHERAPY

6.1 Introduction

High-dose-rate brachytherapy (HDR-BT) involves placing a radiation source inside of or adjacent to a target organ, i.e., tumor. Conventional HDR-BT uses an unshielded brachytherapy source with a radially-symmetric dose distribution [115, 148], which limits the intensity modulation capacity of the approach. Rotating-shield brachytherapy (RSBT) has a rotating radiation-attenuating shield around a brachytherapy source. The RSBT concepts for single-catheter treatment [75] and multi-catheter treatment [76] were introduced by Ebert in 2002 and 2006 respectively.

In the multi-helix RSBT (H-RSBT) treatment, a radiation source travels inside a brachytherapy applicator having helical keyways. While moving along the applicator for a given keyway, the partial shield rotates around the radiation source simultaneously. In traveling along each keyway, the radiation source stops at designated locations called dwell positions. By adjusting the distance between adjacent dwell positions, the rotation angle of the partial shield is determined accordingly. Intensity modulated dose distributions can be delivered to the target with reduced dose exposure to non-target organs by controlling the treatment time in an optimal manner for each dwell position. Hence, it is reported that a radiation source with rotating shields can deliver more conformal dose distributions than an unshielded radiation source. [59]

HDR-BT treatment plans are often generated using inverse planning tools [2, 7, 53, 63, 104, 121, 133]. Based on the given clinical prescription, various optimization problems were introduced previously ranging from minimizing treatment time under restrictions [116, 153] to minimizing dose error [3, 104, 121, 133]. Inverse planning by simulated annealing (IPSA) [121] is a well known method to optimize the dose volume histogram (DVH) directly with given constraints. The BrachyVision treatment planning system (Varian Medical System Inc., Palo Alto, CA) uses this type of DVH-based optimization algorithm. [133]

Unlike the conventional HDR-BT optimization, the RSBT optimization problem has the additional optimization variables of radiation exposure time at each angle of the shield. Due to the increased degrees of freedom in RSBT, RSBT optimization is more difficult than that for the conventional HDR-BT. In addition, there is a compelling need to quickly obtain optimal treatment plans in RSBT to enable clinical usage. To achieve this, researchers have used the dose-surface optimization (DSO) method [126, 193], which minimizes the total dose errors over only voxels on the HR-CTV surface. Instead of dealing with only voxels on the HR-CTV surface, Liu *et al.* [125] defined the region of interest in tumor which includes the surface of HR-CTV as well as the inside voxels of tumor. Additionally, the authors used the total variation (TV) norm penalty in their optimization problem to make smooth changes in the emission times of adjacent beams in the treatment process to facilitate the efficient delivery of an RSBT plan. This optimization problem for RSBT is called asymmetric dose-volume optimization with smoothness control (ADOS).

In this paper, we consider the ADOS optimization problem. A fast computational method is proposed to solve the ADOS optimization problem for the optimal cancer treatment planning for RSBT. Liu *et al.* used a commercial optimization solver called CPLEX [125]. In order to efficiently solve the ADOS optimization problem, which is a large-scale RSBT optimization problem, we designed an optimization method based on the proximal graph solver (POGS) [85], which is a solver using the alternating direction method of multipliers (ADMM). For using POGS, we derived closed-form formulas for the proximal operators used in POGS. Further, we applied our method to the H-RSBT, which is a mechanically-feasible delivery technique for RSBT proposed by Dadkhah *et al.* [59]. In the numerical experiments, we considered cervical cancer, even though our method is also applicable to other types of cancer such as breast cancer and prostate cancer.

6.2 Background on the Multi-helix RSBT

6.2.1 Delivery method

In order to deliver the radiation dose to a target organ, we consider a mechanically feasible delivery technique for RSBT, called the multi-helix RSBT (H-RSBT) [59]. Fig. 6.1 shows the illustration of the H-RSBT method. The shield opening is represented by the *azimuthal* and *zenith* emission angles, denoted by $\Delta\varphi$ and $\Delta\theta$ respectively.

In H-RSBT, a Xoft Axxent radiation source, inside its cooling catheter, with a freely-rotating partial radiation shield is translated inside an applicator with six

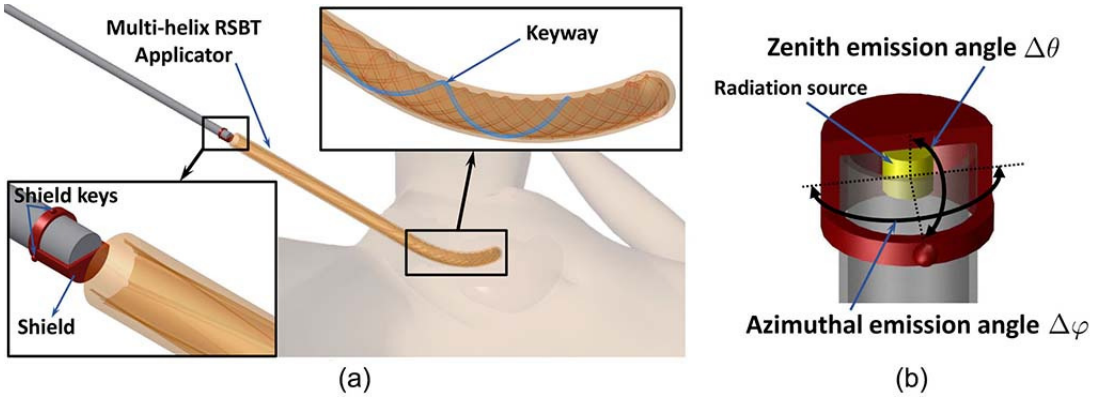


Figure 6.1: (a) Illustration of multihelix rotating shield brachytherapy (H-RSBT) system. (b) Partially shielded radiation source.

helical keyways carved out of the inner wall. The six keyways are evenly spaced on the applicator cross section, by 60°, and each keyway has a helical pitch of one rotation per 33.3 mm of translation. The partial shield has a protruding key that travels down a given keyway, and, due to the helical design of the keyways, the shield rotates about the radiation source as the source catheter is translated, and the emission angle of the shield is known for a given keyway and translational dwell position. As the H-RSBT applicator has 6 helices, with 33.3 mm of translation along the applicator per helical rotation and 1.7 mm spacing between dwell positions, it yields 17.5° of rotation for the shield per 5 mm (standard dwell position spacing) of its translation along the applicator. The dose calculation resolution was $1\times1\times3\ mm^3$ for all cases. The transmission through the shield is 0.1% and approximated to be 0% for the dose calculation. For each patient considered, 45° azimuthal emission angle was used for treatment planning. The zenith emission angle of the modeled shields was an asymmetric 116°, which is consistent with previous work. [59]

6.2.2 Radiation source model and dose calculation

For H-RSBT, the delivery is parameterized by keyway number and dwell position number along the keyway. To quantitatively describe the structure of high-dose-regions formed by a partially shielded source, we introduce the notation of *beamlet*. A beamlet, denoted by $D_i(j,b)$, is defined as the dose rate at point $\vec{r}_i$ with the shielded source positioned at the j-th dwell position while the shield is aligned with the b-th keyway.

To calculate the beamlet, we use the TG-43 dose calculation model of Rivard *et al.* [154]. The radiation source is assumed to be partially shielded 50kVp Xoft Axxent^{TM} (Sunnyvale, CA). To be consistent with previous work [126, 182, 193], we consider that the dose to the points blocked by the shield is 0, since the transmission rate from 50kVp Xoft Axxent^{TM} can be controlled to be less than 0.1% when using a 0.5 mm tungsten shield. [126, 193] Then, we can quantify the radiation dose amount at the point $\vec{r}_i$, denoted by d_i, as a time-weighted sum of all beamlets as follows:

$$d_i = \sum_{j,b} D_i(j,b) t_{j,b}, \tag{6.1}$$

where $t_{j,b}$ is the duration time of the beamlet $D_i(j,b)$.

In the next subsection, we introduce the RSBT optimization problem having asymmetric penalty parameters for HR-CTV and organs at risk (OARs), with TV regulation term for smoothness in the beamlet emission times.

6.3 Optimization problem for cancer treatment planning in RSBT

6.3.1 Problem Formulation

Let $t \in \mathbb{R}^{mn\times 1}$ be the beamlet emission time vector for all keyways and all dwell positions, where m and n are the number of keyways, i.e, $m = 6$, and the number of dwell positions along a keyway respectively. We can obtain t by vectorizing $t_{j,b}$ in Eqn. (6.1); namely, the vector $t \in \mathbb{R}^{mn\times 1}$ is a concatenated vector, which is expressed as $t = [t[1]^T, t[2]^T, ..., t[m]^T]^T$, where $t[j] \in \mathbb{R}^{n\times 1}$ is the dwell time vector for all the beamlets along a keyway, and the super-script T represents the transpose. Let us define a concatenated dose rate matrix $D = [D[1], D[2], ..., D[m]] \in \mathbb{R}^{l\times mn}$, where $D[j] \in \mathbb{R}^{l\times n}$, $j = 1, ..., m$, is the dose rate matrix for the j-th selected keyway, and l is the number of voxels that we are interested in. We denote the whole index set for voxels of interest (VOIs) as $\mathcal{I}_{VOIs}$ and the index set for HR-CTV, bladder, rectum, sigmoid, and normal tissue around the HR-CTV as $\mathcal{I}_{tumor}$, $\mathcal{I}_{bladder}$, $\mathcal{I}_{rectum}$, $\mathcal{I}_{sigmoid}$, and $\mathcal{I}_{normal}$ respectively. The dose rate matrix D has information about dose rate delivered to each tissue point in $\mathcal{I}_{VOIs}$ from each beamlet.

We consider the following RSBT optimization problem having a quadratic objective function with total variation (TV) regulation term for smooth beamlet emission times:

$$\begin{aligned} \underset{t\in\mathbb{R}^{mn}, d\in\mathbb{R}^{l}}{\text{minimize}} \quad & \sum_{i\in\mathcal{I}_{VOIs}} h(d_i) + \sum_{j=1}^{m} \beta \|Lt[j]\|_1 \\ \text{subject to} \quad & Dt = d, \\ & t \geq 0, \end{aligned} \tag{6.2}$$

where $D \in \mathbb{R}^{l \times mn}$ is a dose rate matrix, $t \geq 0$ is the element-wise non-negative emission time, d_i, which is the i-th element of d, is the dose amount at the i-th voxel as introduced in Eqn. (6.1). In Eqn. (6.2),

$$h(d_i) \triangleq (\lambda_i^+ H(d_i - \hat{d}_i) + \lambda_i^- H(\hat{d}_i - d_i))(d_i - \hat{d}_i)^2,$$

where $H(x)$ is the unit step function, which is $H(a) = 1$ if $a > 0$, and $H(a) = 0$ if $a \leq 0$, and λ_i^+ and λ_i^- represent overdose and underdose penalty parameters for the i-th voxel respectively. $L \in \mathbb{R}^{n \times n}$ is the matrix which calculates TV norm of a vector; namely, L is defined as follows:

$$L \triangleq \begin{bmatrix} 1 & -1 & 0 & 0 & \dots & 0 & 0 \\ 0 & 1 & -1 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & -1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & 1 & -1 \\ 0 & 0 & 0 & 0 & \dots & 0 & 0 \end{bmatrix}.$$

$\hat{d}_i$ is a prescribed dose amount for the i-th voxel. $\hat{d}_i$ can have a different value for each VOI. For example, $\hat{d}_i = \hat{d}_{tumor}$ if $i \in \mathcal{I}_{tumor}$, and $\hat{d}_i = \hat{d}_{badder}$ if $i \in \mathcal{I}_{bladder}$. We denote the prescribed dose amount for HR-CTV, bladder, rectum, sigmoid, and normal tissue around the HR-CTV as $\hat{d}_{tumor}$, $\hat{d}_{bladder}$, $\hat{d}_{rectum}$, $\hat{d}_{sigmoid}$, and $\hat{d}_{normal}$. For λ_i^+ and λ_i^-, we use different non-negative overdose and underdose parameter values including 0.

The TV norm alleviates the positioning uncertainty in the treatment process. If we have two dramatically different emission times $t_{j,b}$ and $t_{j+1,b}$ between two adjacent beamlets along the same keyway, a small error in the dwell positions may cause

an unacceptable treatment result. By applying the smoothness term between two adjacent beamlets along the same keyway in H-RSBT, we can reduce the treatment error caused by the positioning uncertainty in the treatment process.

Since we use different penalty parameter values for the overdose and underdose of a voxel, we call Eqn. (6.2) as RSBT optimization problem having asymmetric penalty parameters or simply ADOS.

6.3.2 POGS implementation

In order to simplify the sum of the TV norms in Eqn. (6.2), let us introduce a matrix $\bar{L} \triangleq I_{m\times m} \otimes L$, where $\otimes$ is the Kronecker product, and $I_{m\times m}$ is an $m \times m$ identity matrix. By assigning $\bar{L}t = y$ and introducing the indicator function $I(\cdot)$, we restate Eqn. (6.2) as follows:

$$\begin{aligned} \underset{t,y\in\mathbb{R}^{mn},d\in\mathbb{R}^{l}}{\text{minimize}} \quad & \sum_{i\in\mathcal{I}_{VOIs}} h(d_i) + \beta\|y\|_1 + I(x \geq 0) \\ \text{subject to} \quad & Dt = d, \\ & \bar{L}t = y, \end{aligned} \tag{6.3}$$

where $D \in \mathbb{R}^{l\times mn}$, $\bar{L} \in \mathbb{R}^{mn\times mn}$, and $I(t \geq 0)$ is the element-wise indicator function; namely, $I(t_i \geq 0) = 0$ if $t_i \geq 0$, and $I(t_i \geq 0) = \infty$ if $t_i < 0$.

By letting

$$A = \begin{pmatrix} D \\ \bar{L} \end{pmatrix}, \quad z = \begin{pmatrix} d \\ y \end{pmatrix},$$

we can further simplify Eqn. (6.3) into

$$\begin{aligned} \underset{t\in\mathbb{R}^{mn}, z\in\mathbb{R}^{l+mn}}{\text{minimize}} \quad & \sum_{i=1}^{l} h(z_i) + \beta\|z_{[l+1:l+mn]}\|_1 + I(t \geq 0) \\ \text{subject to} \quad & At = z, \end{aligned} \tag{6.4}$$

where $z_{[a:b]}$ is the partial vector of z by taking vector z from the a-th element to the b-th element. We define the following functions:

$$g(t) = I(t \geq 0), \tag{6.5}$$

$$f(z) = \sum_{i=1}^{l} h(z_i) + \beta\|z_{[l+1:l+mn]}\|_1. \tag{6.6}$$

Then we turn Eqn. (6.4) into a graph-form convex optimization problem [85], where the constraint is $z = At$, and $A = [D^T \bar{L}^T]^T \in \mathbb{R}^{(l+mn)\times mn}$. We have derived the detailed updating rules for each optimization variable in the POGS solver for Eqn. (6.4). We introduce our derived results in detail for the proximal operators used in the POGS solver, updating steps, and stopping criteria as follows.

The POGS updates primal variables, conducts the projection onto the space $z = At$, and then, updates dual variables iteratively until the stopping criteria are satisfied or the maximum number of iterations, denoted by $MaxItr$, is reached. The primal variable and dual variable are updating variables to be used for optimality condition in the algorithm. For the primal variable, dual variable, and projection result, we used $(\hat{t}, \hat{z})$, (t, z), and $(\tilde{t}, \tilde{z})$ respectively. We introduce each updating steps in detail for our optimization problem in Eqn. (6.4). We use the super-script k to represent the k-th iteration.

Updating primal variables $\hat{t}^{k+1}$ and $\hat{z}^{k+1}$: In updating the primal variables, we use the following proximal operators with a penalty parameter ρ:

$$\begin{aligned}
\hat{t}^{k+1} &= Prox_g(t^k - \tilde{t}^k) \\
&= \underset{t}{argmin}\ \ I(t \geq 0) + \frac{\rho}{2}||t - (t^k - \tilde{t}^k)||^2, \\
\hat{z}^{k+1} &= Prox_f(z^k - \tilde{z}^k) \\
&= \underset{z}{argmin}\ \ \sum_{i=1}^{l} h(z_i) + \beta||z_{[l+1:l+mn]}||_1 + \frac{\rho}{2}||z - (z^k - \tilde{z}^k)||^2.
\end{aligned}$$

The proximal operator is used to make a compromise between the solution at the k-th iteration and the function value with the solution at the $k+1$ iteration. We are able to explicitly derive closed-form formulas for the proximal operators. For $\hat{t}^{k+1} \in \mathbb{R}^{mn}$, we have

$$\hat{t}^{k+1} = max(t^k - \tilde{t}^k, \mathbf{0}), \tag{6.7}$$

where $max(a, b)$ provides the maximum value between a and b element-wise. For $\hat{z}_i^{k+1}$, $1 \leq i \leq l$, we also derive

$$\hat{z}_i^{k+1} = \begin{cases} z_i^k - \tilde{z}_i^k - \frac{\beta}{\rho}, & \text{if } z_i^k - \tilde{z}_i^k > \frac{\beta}{\rho} \\ z_i^k - \tilde{z}_i^k + \frac{\beta}{\rho}, & \text{if } z_i^k - \tilde{z}_i^k < -\frac{\beta}{\rho} \\ 0, & \text{otherwise} \end{cases}. \tag{6.8}$$

For $\hat{z}_i^{k+1}$, $l+1 \leq i \leq l+mn$, we obtain

$$\hat{z}_i^{k+1} = \begin{cases} \frac{2\lambda_i^+ \hat{d}_i + \rho(z_i^k - \tilde{z}_i^k)}{2\lambda_i^+ + \rho}, & \text{if } z_i^k - \tilde{z}_i^k \geq \hat{d}_i, \\ \frac{2\lambda_i^- \hat{d}_i + \rho(z_i^k - \tilde{z}_i^k)}{2\lambda_i^- + \rho}, & \text{if } z_i^k - \tilde{z}_i^k < \hat{d}_i. \end{cases} \tag{6.9}$$

POGS uses the adaptive value for ρ as default to further increase the convergence speed.

Projection onto $z = At$ from $(\hat{t}^{k+1}+\tilde{t}^{k}, \hat{z}^{k+1}+\tilde{z}^{k})$: The projection operation is mapping the primal variables to the closest feasible solution. The projected variables onto $z = At$ from $(\hat{t}^{k+1}+\tilde{t}^{k}, \hat{z}^{k+1}+\tilde{z}^{k})$, denoted as t^{k+1} and z^{k+1}, are obtained by solving the following optimization:

$$\begin{aligned} \underset{t,z}{\text{minimize}} \quad & \frac{1}{2}\|t - (\hat{t}^{k+1} + \tilde{t}^{k})\|_2^2 + \frac{1}{2}\|z - (\hat{z}^{k+1} + \tilde{z}^{k})\|_2^2 \\ \text{subject to} \quad & At = z \end{aligned}$$

By solving this optimization and using Lagrange conditions [20], we have the following formulation:

$$\begin{pmatrix} t^{k+1} \\ z^{k+1} \end{pmatrix} = \begin{pmatrix} I & A^T \\ A & -I \end{pmatrix}^{-1} \begin{pmatrix} \hat{t}^{k+1} + \tilde{t}^{k} + A^T(\hat{z}^{k+1} + \tilde{z}^{t}) \\ 0 \end{pmatrix}.$$

Updating dual variables $\tilde{t}^{k+1}$ and $\tilde{z}^{k+1}$: We obtain the dual variable at iteration $(k+1)$ by updating the dual variable at the k-th iteration as follows:

$$\tilde{t}^{k+1} = \tilde{t}^{k} + \hat{t}^{k+1} - t^{k+1},$$

$$\tilde{z}^{k+1} = \tilde{z}^{k} + \hat{z}^{k+1} - z^{k+1}.$$

We summarize the updating steps in Algorithm 6.1.

Stopping criteria: For the stopping criteria, we define the primal and dual residuals as follows:

$$\begin{aligned} &\|A\hat{t}^{k+1} - \hat{z}^{k+1}\|_2 \le \epsilon^{pri}, \\ &\|A^T\hat{v}^{k+1} + \hat{\mu}^{k+1}\|_2 \le \epsilon^{dual}, \end{aligned} \tag{6.10}$$

Algorithm 6.1 Fast treatment planning for RSBT in POGS implementation

Input: $A \in \mathbb{R}^{(l+mn)\times mn}$, $MaxItr$, $\lambda^+, \lambda^-, \hat{d} \in \mathbb{R}^l$, β
Output: t
Initialize: $k \leftarrow 0$, $t^k \leftarrow 0$, $z^k \leftarrow 0$, $\tilde{t} \leftarrow 0$,$\tilde{z} \leftarrow 0$
for $k = 1$ **to** $MaxItr$ **do**
  Updating primal variables $\hat{t}^{k+1}$, $\hat{z}^{k+1}$:
  $\hat{t}^{k+1} \leftarrow Prox_g(t^k - \tilde{t}^k)$ ▷ See (6.7)
  $\hat{z}^{k+1} \leftarrow Prox_f(z^k - \tilde{z}^k)$ ▷ See (6.8) and (6.9)
  Projection onto $z = At$:

$$\begin{pmatrix} t^{k+1} \\ z^{k+1} \end{pmatrix} \leftarrow \begin{pmatrix} I & A^T \\ A & -I \end{pmatrix}^{-1} \begin{pmatrix} \hat{t}^{k+1} + \tilde{t}^k + A^T(\hat{z}^{k+1} + \tilde{z}^t) \\ 0 \end{pmatrix}$$

  Updating dual variables $\tilde{t}^{k+1}$, $\tilde{z}^{k+1}$:
  $\tilde{t}^{k+1} \leftarrow \tilde{t}^k + \hat{t}^{k+1} - t^{k+1}$
  $\tilde{z}^{k+1} \leftarrow \tilde{z}^k + \hat{z}^{k+1} - z^{k+1}$
  if *Stopping criteria are met* **then**
    break
  end
end

where $\hat{v}^{k+1} = -\rho(\hat{z}^{k+1} - z^k + \tilde{z}^k)$, $\hat{\mu}^{k+1} = -\rho(\hat{t}^{k+1} - t^k + \tilde{t}^k)$. Here, ϵ^{pri} and ϵ^{dual} are positive tolerances for primal and dual residuals respectively:

$$\epsilon^{pri} = \epsilon^{abs} + \epsilon^{rel}||\hat{z}^{k+1}||_2,$$
$$\epsilon^{dual} = \epsilon^{abs} + \epsilon^{rel}||\hat{\mu}^{k+1}||_2, \tag{6.11}$$

where we used $(\epsilon^{abs}, \epsilon^{rel}) = (10^{-4}, 10^{-2})$ in the numerical experiments.

6.4 Treatment Planning

Five patients with cervical cancer were considered, whose HR-CTV volumes range from 42.2 to 98.8 cm^3. Table 6.1 shows the volume and maximum dimension of HR-CTV for five patients. All the HR-CTVs and OARs were manually contoured by physicians on T2 weighted $1 \times 1 \times 3$ mm^3 resolution MR images taken with a Siemens MAGNETOM 3T scanner (Siemens, Germany) at the beginning of the first fraction

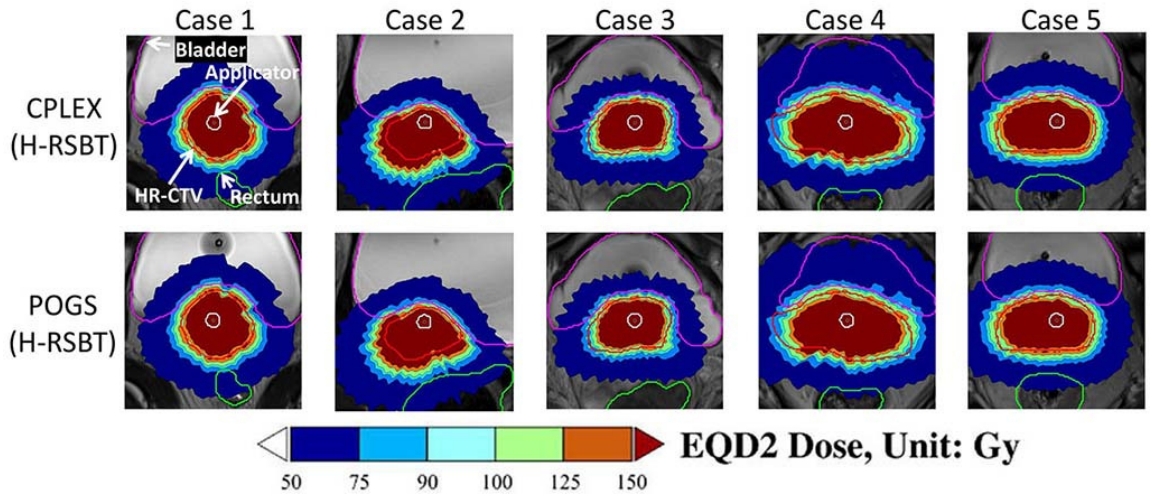


Figure 6.2: EQD2 dose distributions on MR images for five patient cases obtained from CPLEX and POGS with H-RSBT using 45° azimuthal angle.

of brachytherapy. A titanium Fletcher-Suit-Delclos style tandem and ovoids (Varian Medical Systems, Palo Alto, CA) were used as the brachytherapy applicator. We used the same datasets as the previous research conducted by Liu *et al.* [125] and Dadkhah *et al.* [59].

Table 6.1: HR-CTV volumes and dimensions for all patients

Patient Num.	HR-CTV volume (cm^3)	HR-CTV maximum dimension (cm)
Case 1	42.2	6.3
Case 2	45.8	7.4
Case 3	78.0	8.6
Case 4	98.8	9.6
Case 5	75.0	7.5
Avg.	68.0	7.9
SD[a]	23.8	1.8

[a] Standard Deviation

All the patients had external beam radiation treatment in 25 fractions at 1.8

Table 6.2: Parameter settings

Method	$\hat{d}_{tumor}$	$\hat{d}_{bladder}$	$\hat{d}_{rectum}$	$\hat{d}_{sigmoid}$	$\hat{d}_{normal}$ [a]	Tumor $\lambda_i^+/\ \lambda_i^-$	Bladder $\lambda_i^+/\ \lambda_i^-$	Rectum $\lambda_i^+/\ \lambda_i^-$	Sigmoid $\lambda_i^+/\ \lambda_i^-$	Normal [b] $\lambda_i^+/\ \lambda_i^-$	β
CPLEX	40	25	20	20	40	0/ 2	2/ 0	2/ 0	2/ 0	2/ 0	100
POGS	40	25	20	20	40	0/ 2	2/ 0	2/ 0	2/ 0	2/ 0	100

[a] Prescribed dose amount for tumor boundary [b] Penalty parameter for tumor boundary

Table 6.3: Comparison between POGS and CPLEX for 45° azimuthal angle

Case	Method	HR-CTV D_{90} (Gy)	HR-CTV D_{100} (Gy)	Bladder D_{2cc} (Gy)	Rectum D_{2cc} (Gy)	Sigmoid D_{2cc} (Gy)	Execution time (sec.)
Case 1	CPLEX	110.8	54.0	89.9	62.4	75.0	32.1
	POGS	111.4	54.0	90.0	64.7	74.7	2.1
Case 2	CPLEX	111.5	44.3	90.0	72.2	54.4	37.0
	POGS	111.5	44.3	90.0	71.7	54.8	2.1
Case 3	CPLEX	96.0	44.3	85.9	57.3	75.0	65.4
	POGS	95.0	44.3	85.2	55.1	75.0	3.9
Case 4	CPLEX	107.0	55.3	90.0	69.9	54.0	39.4
	POGS	106.9	55.4	90.0	69.8	54.0	2.3
Case 5	CPLEX	112.7	44.3	90.0	68.1	59.2	65.4
	POGS	112.7	44.3	90.0	68.1	59.2	3.2
Average	CPLEX	107.6	48.4	89.2	66.0	63.5	47.9
	POGS	107.5	48.5	89.0	65.9	63.5	2.7

Gy/fraction. We assume that the external beam radiotherapy dose was uniformly delivered to the HR-CTV and OARs for all the patient cases. The dose in each voxel was converted to equivalent dose in 2 Gy per fraction of external radiation therapy (EQD2) using the linear quadratic model, [109] where the linear-quadratic parameter, α/β, is set to 3 Gy for OARs and 10 Gy for HR-CTV.

For VOIs, we define the voxels located at a distance between 3 mm and 20 mm to the radiation source path or those within 10 mm inside and outside of the HR-CTV boundary surface. [125] We deal with the HR-CTV, HR-CTV boundary, bladder, sigmoid, and rectum inside of VOIs in our optimization problem. The optimization

parameter settings are shown in Table 6.2.

For all the brachytherapy treatment plans, we escalated the EQD2 of the HR-CTV without exceeding the D_{2cc} tolerance of the bladder, rectum, and sigmoid colon. We used 90 Gy for bladder tolerance, and 75 Gy for rectum and sigmoid colon tolerances according to Groupe Européen de Curiethérapie, European Society for Therapeutic Radiology and Oncology (GEC ESTRO). [92, 149]

6.5 Numerical Experiments

Optimized treatment plans were generated for all patients using the POGS method and the previously considered CPLEX method. [125] The same objective function, with the same input parameters and beamlets, was minimized for each patient with both methods. A total variation term was included in the objective function as a regularization term, resulting in smoothly-varying emission times along each keyway. The regularization promotes robustness of the resulting overall dose distribution with respect to small errors (expected ≤ 1 mm) in source positioning. The rectum, bladder, sigmoid colon, HR-CTV, and HR-CTV boundary were the structures considered.

We compared our method with the previous research conducted by Liu *et al.* [125] using CPLEX [103] for H-RSBT. We evaluated the quality of the delivery plans as well as the execution time to solve Eqn. (6.2) with POGS [85]. Since Liu *et al.* compared their method based on CPLEX with other existing RSBT dose optimization methods ranging from DSO to IPSA in their previous research [125], we

only compared POGS and CPLEX in this paper.

The comparison metrics for the quality of the delivery plans are the HR-CTV D_{90}, HR-CTV D_{100}, OARs D_{2cc}, DVH, and dose distributions. Since the goal of this research is achieving a fast solution to the RSBT dose optimization problem without compromising the plan quality, we compared the execution times to solve Eqn. (6.2) for all five patient cases. We conducted our numerical experiments on HP Z220 CMT with an Intel Core i7-3770 dual core CPU @3.4GHz clock speed and 16GB DDR3 RAM, using Matlab (R2013b) on the Windows 7 operating system.

Table 6.4: Dimension of $D \in \mathbb{R}^{l \times mn}$ in Eqn. (6.2)

Patient Num.	$l \times mn$
Case 1	54693×144
Case 2	51109×126
Case 3	79065×222
Case 4	50680×144
Case 5	59220×228
Avg.	58953×173

Results for five cervical cancer patient cases are shown in Table 6.3. Table 6.4 shows the dimension of the dose matrix D in Eqn. (6.2). Fig. 6.3 shows the corresponding DVH in H-RSBT. With the same parameter settings as in Table 6.2, POGS can achieve an RSBT plan with almost the same D_{90} value (less than 1% difference) as that achieved by CPLEX in each of the five patient cases. D_{2cc} values for OARs obtained by POGS are also almost the same (less than 1% difference) as those from CPLEX. For the execution time, we achieved around 18 times faster speed to solve

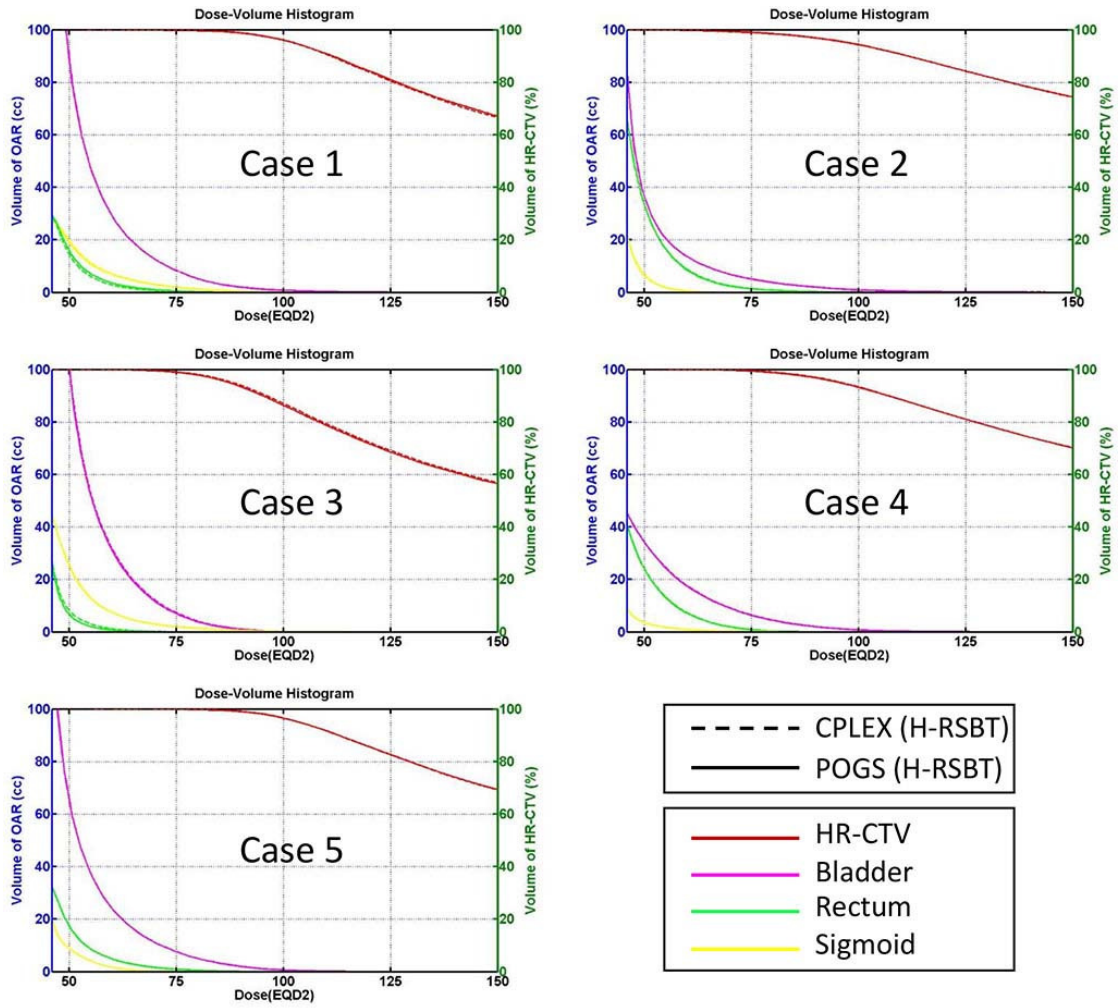


Figure 6.3: Dose volumn histograms (DVH) of all treatment planning for five patient cases in H-RSBT with 45° azimuthal angle.

the ADOS problem for H-RSBT than the CPLEX based method on average. Over all patients, total optimization times were 32.1-65.4 seconds for CPLEX and 2.1-3.9 seconds for POGS.

Fig. 6.2 shows that the EQD2 figures were similar for each case between CPLEX and POGS.

6.6 Discussion

Various treatment planning methods in radiation therapy have been studied. One of the well known methods, which is called IPSA, was introduced to directly optimize DVH with given constraints in heuristic way. Due to its heuristic nature, a global solution is not guaranteed. Instead of directly optimizing DVH, we consider the voxel-wise optimization problem for the RSBT treatment planning, called ADOS optimization problem, which can be expressed in a convex optimization problem. We can take advantage of the convexity to obtain a global solution.

In the ADOS optimization problem, we reduced the size of the ADOS optimization problem by defining VOIs. Instead of defining VOIs, the whole voxels can be considered in the optimization problem under the expectation of better treatment quality with heavy computation. Since parallel computing and GPU-based high performance computing can play an important role in solving extremely large-scale optimization problems, there is a rising question about the usability of POGS in parallel computing environment or GPU-based implantation. The implementation of POGS in such environment is another research work.

In addition, we used a partial shield with 4° azimuthal angle in H-RSBT for our numerical experiments. However, finding the optimal shield angle in H-RSBT is still an open problem. In order to determine the size of angle and the radiation exposure time at each angle of shield, considering both variables in the optimization problem is also a possible optimization problem.

Finally, POGS (and ADMM) was used in previous research on intensity-

modulated radiation therapy (IMRT) [85, 198], fluence map optimization [86], and external beam radiotherapy (EBRT) optimization [124]. Right at the time of submitting our journal manuscript, we learned of the recently-appearing (published on April 12th, 2017) work [124] which applied POGS algorithms to EBRT dose optimization. The work [124] focused on EBRT, while our paper is the first work to use POGS in brachytherapy, including the mechanically-feasible delivery system called H-RSBT. In our paper, we use TV norm to promote smoothly-varying emission times along each keyway, such that the treatment plan is robust to positioning errors of dwell positions. By comparison, the TV norm is instead applied to promote the smoothness of the resulting fluence map in the research [124] and simplify the delivery. Our proposed method is applicable to conventional HDR-BT as well as dynamic modulated brachytherapy [183] with simple modifications, since they share similar mechanisms as H-RSBT.

6.7 Conclusion

POGS substantially reduced treatment plan optimization time around 18 times for RSBT with similar HR-CTV D_{90}, OAR D_{2cc} values, and EQD2 figure comparing to CPLEX, which is significant progress toward clinical translation of RSBT. Over all cervical cancer patients, total optimization times were 32.1-65.4 seconds for CPLEX and 2.1-3.9 seconds for POGS. POGS is also applicable to conventional high-dose-rate brachytherapy.

CHAPTER 7
CONCLUSIONS AND FUTURE WORK

Our future plans for each subject are provided in the following sections.

7.1 Super-resolution

Based on our previous research on super-resolution, applying our proposed super-resolution theories and algorithms to various applications can be one specific research area. The possible applications include X-ray crystallography, optic microscopy, and Direction of Arrival (DoA) radar. Since images can be understood as the sum of unit step functions, by calculating the derivative of images, we can express the images as the sum of Dirac delta functions; our super-resolution algorithms ranging from the phaseless super-resolution algorithm to the super-resolution with prior information are then applicable. Therefore, by using our super-resolution methods, we can obtain high resolution images with low frequency magnitude information, which is one specific example. Designing numerical algorithms to solve even larger super-resolution problems in 2D or 3D fast is another direction of this research.

7.2 Designing algorithms for distributed system

As introduced in the introduction and motivation section, optimization problems in the technological era of ubiquitous computing and networking will be even larger and much more complicated than the traditional optimization problems. In addition, the amount of data that we can access is skyrocketing. Therefore, the de-

mand for efficient algorithms to solve the large scale optimization problems in the distributed system model is extensively increasing. As for designing algorithms for distributed systems, current research involves reducing the complexity of the optimization problems by taking advantage of data structure information, as well as dealing with distributed processors and memories. For this research, use of the first order methods such as coordinate descent, stochastic gradient descent, and dual coordinate ascent has been encouraged due to their low complexity and adequate quality of solution. The communication-efficient first order method is especially interesting for this research. The authors in [106] proposed the distributed dual coordinate ascent method with the geometric convergence rate on the optimization problem minimizing a convex and smooth loss function with a convex regularization term in a star network model. In order to deal with highly connected system models targeting the Internet of Things (IoTs) and big data problems, designing general algorithms for various network models such as ring, randomly connected models, and fully connected models is one potential research topic [50].

7.3 Optimal treatment plan for rotating-shield brachytherapy

In the rotating-shield brachytherapy for prostate cancer, around 15-20 needles are used to deliver radiation source to prostate. In the previous research [1], the authors have chosen the locations of needles somehow arbitrary. For optimal treatment planning, determining the optimal needle locations as well as the optimal number of needles is an open problem. Hence, we can conduct research on developing an

algorithm for prostate cancer to choose the optimal number of needles as well as the locations. For the possible needle locations, we are able to discretize the target organ and consider each grid point as a possible needle location. In the algorithm, we can consider block sparsity to choose possible needle locations, since tumor tissues can be located closely each other. Therefore, designing an algorithm for prostate is a possible direction for the rotating-shield brachytherapy.

REFERENCES

[1] Q. E. Adams, J. Xu, E. K. Breitbach, X. Li, S. A. Enger, W. R. Rockey, Y. Kim, X. Wu, and R. T. Flynn. Interstitial rotating shield brachytherapy for prostate cancer. *Medical physics*, 41(5):051703, 2014.

[2] T. Akimoto, H. Katoh, Y. Kitamoto, K. Shirai, M. Shioya, and T. Nakano. Anatomy-based inverse optimization in high-dose-rate brachytherapy combined with hypofractionated external beam radiotherapy for localized prostate cancer: comparison of incidence of acute genitourinary toxicity between anatomy-based inverse optimization and geometric optimization. *International Journal of Radiation Oncology, Biology, Physics*, 64(5):1360–1366, 2006.

[3] R. Alterovitz, E. Lessard, J. Pouliot, I.-C. J. Hsu, J. F. OBrien, and K. Goldberg. Optimization of hdr brachytherapy dose distributions using linear programming with penalty costs. *Medical Physics*, 33(11):4012–4019, 2006.

[4] D. Angelosante, G. Giannakis, and E. Grossi. Compressed sensing of time-varying signals. In *Proceedings of IEEE International Conference on Digital Signal Processing*, pages 1–8, 2009.

[5] H. Attouch, J. Bolte, P. Redont, and A. Soubeyran. Proximal alternating minimization and projection methods for nonconvex problems: An approach based on the kurdyka-lojasiewicz inequality. *Mathematics of Operations Research*, 35(2):438–457, 2010.

[6] J.-M. Azais, Y. De Castro, and F. Gamboa. Spike detection from inaccurate samplings. *Applied and Computational Harmonic Analysis*, 38(2):177–195, 2015.

[7] D. Baltas, Z. Katsilieri, V. Kefala, S. Papaioannou, A. Karabis, P. Mavroidis, and N. Zamboglou. Influence of modulation restriction in inverse optimization with hipo of prostate implants on plan quality: analysis using dosimetric and radiobiological indices. In *Proceedings of World Congress on Medical Physics and Biomedical Engineering*, pages 283–286. Springer, 2009.

[8] R. G. Baraniuk, V. Cevher, M. F. Duarte, and C. Hegde. Model-based compressive sensing. *IEEE Transactions on Information Theory*, 56(4):1982–2001, 2010.

[9] M. Basseville and I. Nikiforov. *Detection of Abrupt Changes: Theory and Application*. Prentice-Hall, 1993.

[10] A. Beck and M. Teboulle. A fast iterative shrinkage thresholding algoirthm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009.

[11] H. Benveniste and S. Blackband. MR microscopy and high resolution small animal MRI: applications in neuroscience research. *Progress in neurobiology*, 67(5):393–420, 2002.

[12] R. Berinde, A. Gilbert, P. Indyk, H. Karloff, and M. Strauss. Combining geometry and combinatorics: a unified approach to sparse signal recovery. In *Proceedings of Allerton conference on Communication, Control, and Computing*, pages 798–805, 2008.

[13] S. Beygi and U. Mitra. Multi-scale multi-lag channel estimation using low rank structure of received signal. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4299–4303, 2014.

[14] T. Blu, P.-L. Dragotti, M. Vetterli, P. Marziliano, and L. Coulot. Sparse sampling of signal innovations. *IEEE Signal Processing Magazine*, 25(2):31–40, 2008.

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

[16] J. Bolte, S. Sabach, and M. Teboulle. Proximal alternating linearized minimization for nonconvex and nonsmooth problems. *Mathematical Programming*, 146(1-2):459–494, 2014.

[17] S. Bourguignon, H. Carfantan, and J. Idier. A sparsity-based method for the estimation of spectral lines from irregularly sampled data. *IEEE Journal on Selected Topics in Signal Processing*, 1(4):575–585, 2007.

[18] N. Boyd, G. Schiebinger, and B. Recht. The alternating descent conditional gradient method for sparse inverse problems. *arXiv preprint arXiv:1507.01562*, 2015.

[19] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.

[20] S. P. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge University Press, 2004.

[21] S. Brenner and R. W. Horne. A negative staining method for high resolution electron microscopy of viruses. *Biochimica et biophysica acta*, 34:103–110, 1959.

[22] J.-F Cai, E. J. Candès, and Z. Shen. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20(4):1956–1982, 2010.

[23] J.-F. Cai, S. Liu, and W. Xu. Projected Wirtinger gradient descent for low-rank hankel matrix completion in spectral compressed sensing. *arXiv preprint arXiv:1507.03707*, 2015.

[24] E. J. Candes, Y. C. Eldar, T. Strohmer, and V. Voroninski. Phase retrieval via matrix completion. *SIAM Journal on Imaging Sciences*, 6(1):199–225, 2013.

[25] E. J. Candès, Y. C. Eldar, T. Strohmer, and V. Voroninski. Phase retrieval via matrix completion. *SIAM review*, 57(2):225–251, 2015.

[26] E. J. Candès and C. Fernandez-Granda. Super-resolution from noisy data. *Journal of Fourier Analysis and Applications*, 19(6):1229–1254, 2013.

[27] E. J. Candès and C. Fernandez-Granda. Towards a mathematical theory of super-resolution. *Communications on Pure and Applied Mathematics*, 67(6):906–956, 2014.

[28] E. J. Candès and B. Recht. Exact matrix completion via convex optimization. *Foundations of Computational mathematics*, 9(6):717–772, 2009.

[29] E. J. Candès, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, 52(2):489–509, 2006.

[30] E. J. Candès, J. Romberg, and T. Tao. Stable signal recovery from incomplete and inaccurate measurements. *Communications on pure and applied mathematics*, 59(8):1207–1223, 2006.

[31] E. J. Candès and T. Tao. Decoding by linear programming. *IEEE Transactions on Information Theory*, 51(12):4203–4215, 2005.

[32] E. J. Candès, M. B. Wakin, and S. P. Boyd. Enhancing sparsity by reweighted ℓ_1 minimization. *Journal of Fourier analysis and applications*, 14(5-6):877–905, 2008.

[33] C. C. Carathèodory. Über ber den variabilitätsbereich der fourierschen konstanten von positiven harmonischen funktionen. *Rendiconti del Circolo Matematico di Palermo (1884-1940)*, 32(1):193–217, 1911. in German.

[34] C. C. Carathéodory and L. Fejér. Über ber den zusammenhang der extremen von harmonischen funktionen mit ihren koeffizienten und über den picard-landauschen satz. *Rendiconti del Circolo Matematico di Palermo (1884-1940)*, 32(1):218–239, 1911.

[35] R. Caromi, Y. Xin, and L. Lai. Fast multi-band spectrum scanning for cognitive radio systems. *IEEE Transactions on Communications*, 61(1):63–75, 2013.

[36] R. Castro, M. Coates, G. Liang, R. D. Nowak, and B. Yu. Network tomography: recent developments. *Statistical Science*, pages 499–517, 2004.

[37] V. Cevher, M. F. Duarte, C. Hegde, and R. G. Baraniuk. Sparse signal recovery using Markov random fields. In *Proceedings of Advances in Neural Information Processing Systems*, pages 257–264, 2008.

[38] V. Cevher, P. Indyk, C. Hegde, and R. G. Baraniuk. Recovery of clustered sparse signals from compressive measurements. In *Proceedings of International Conference on Sampling Theory and Applications (SAMPTA)*, 2009.

[39] V. Chandrasekaran, B. Recht, P. A. Parrilo, and A. S. Willsky. The convex geometry of linear inverse problems. *Foundations of Computational Mathematics*, 12(6):805–849, 2012.

[40] R. Chartrand and W. Yin. Iteratively reweighted algorithms for compressive sensing. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3869–3872, 2008.

[41] J. Chen, R. Monga, S. Bengio, and R. Jozefowicz. Revisiting distributed synchronous SGD. *arXiv preprint arXiv:1604.00981*, 2016.

[42] M. Chen, S. Mao, and Y. Liu. Big data: A survey. *Mobile Networks and Applications*, 19(2):171–209, 2014.

[43] S. Chen and D. L. Donoho. Examples of basis pursuit. In *Proceedings of Wavelet Applications in Signal and Image Processing III*, pages 564–574, 1995.

[44] S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM journal on scientific computing*, 20(1):33–61, 1998.

[45] Y. Chen and Y. Chi. Robust spectral compressed sensing via structured matrix completion. *IEEE Transactions on Information Theory*, 60(10):6576–6601, 2014.

[46] Y. Chen and Y. Chi. Robust spectral compressed sensing via structured matrix completion. *IEEE Transactions on Information Theory*, 60:6576–5501, 2014.

[47] Y. Chen, Y. C. Eldar, and A. J. Goldsmith. An algorithm for exact super-resolution and phase retrieval. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 754–758, 2014.

[48] Y. Chi and Y. Chen. Compressive recovery of 2-D off-grid frequencies. In *Proceedings of Asilomar Conference on Signals, Systems, and Computers*, pages 687–691, 2013.

[49] Y. Chi, L. L. Scharf, A. Pezeshki, and A. R. Calderbank. Sensitivity to basis mismatch in compressed sensing. *IEEE Transactions on Signal Processing*, 59(5):2182–2195, 2011.

[50] M. Cho, L. Lai, and W. Xu. Network constrained distributed dual coordinate ascent for machine learning. *arXiv preprint arXiv:1703.04785*, 2017.

[51] M. Cho, K. V. Mishra, J.-F. Cai, and W. Xu. Block iterative reweighted algorithms for super-resolution of spectrally sparse signals. *IEEE Signal Processing Letters*, 22(12):2319–2323, 2015.

[52] M. Cho and W. Xu. New algorithms for verifying the null space conditions in compressed sensing. In *Proceedings of Asilomar Conference on Signals, Systems and Computers*, pages 1038–1042, 2013.

[53] D. Citrin, H. Ning, P. Guion, G. Li, R. C. Susil, R. W. Miller, E. Lessard, J. Pouliot, X. Huchen, J. Capala, et al. Inverse treatment planning based on MRI for HDR prostate brachytherapy. *International Journal of Radiation Oncology, Biology, Physics*, 61(4):1267–1275, 2005.

[54] M. J. Coates and R. D. Nowak. Network tomography for internal delay estimation. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 3409–3412, 2001.

[55] A. Cohen, W. Dahmen, and R. DeVore. Compressed sensing and best k-term approximation. *Journal of the American Mathematical Society*, 22(1):211–231, 2009.

[56] C. E. Cook and M. Bernfeld. *Radar signals: an introduction to theory and application*. Artech Print on Demand, 1993.

[57] T. H. Cormen. *Introduction to algorithms*. MIT press, 2009.

[58] T. Cover and J. Thomas. *Elements of Information Theory, 2nd Edition*. Wiley, 2006.

[59] H. Dadkhah, Y. Kim, X. Wu, and R. T. Flynn. Multihelix rotating shield brachytherapy for cervical cancer. *Medical Physics*, 42(11):6579–6588, 2015.

[60] A. d'Aspremont and L. El Ghaoui. Testing the nullspace property using semidefinite programming. *Mathematical programming*, 127(1):123–144, 2011.

[61] R. A. Demillo and R. J. Lipton. A probabilistic remark on algebraic program testing. *Information Processing Letters*, 7(4):193–195, 1978.

[62] W. Deng and W. Yin. On the global and linear convergence of the generalized alternating direction method of multipliers. *Journal of Scientific Computing*, pages 1–28, 2015.

[63] K. D. DeWitt, I. C. J. Hsu, J. Speight, V. K. Weinberg, E. Lessard, and J. Pouliot. 3d inverse treatment planning for the tandem and ovoid applicator in cervical cancer. *International Journal of Radiation Oncology, Biology, Physics*, 63(4):1270–1274, 2005.

[64] D. Donoho. Neighborly polytopes and sparse solution of underdetermined linear equations. *Technical report (Stanford University. Dept. of Statistics)*, 2005.

[65] D. L Donoho. De-noising by soft-thresholding. *IEEE Transactions on Information Theory*, 41(3):613–627, 1995.

[66] D. L. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1306, 2006.

[67] D. L. Donoho. High-dimensional centrally symmetric polytopes with neighborliness proportional to dimension. *Discrete & Computational Geometry*, 35(4):617–652, 2006.

[68] D. L. Donoho and J. Tanner. Neighborliness of randomly projected simplices in high dimensions. In *Proceedings of the National Academy of Sciences of the United States of America*, pages 9452–9457, 2005.

[69] D. L. Donoho and J. Tanner. Precise undersampling theorems. *Proceedings of the IEEE*, 98(6):913–924, 2010.

[70] R. J. Doviak and D. S. Zrnić. *Doppler radar and weather observations*. Dover Publications, second edition, 2006.

[71] M. F. Duarte and R. G. Baraniuk. Spectral compressive sensing. *Applied and Computational Harmonic Analysis*, 35(1):111–129, 2013.

[72] M. F. Duarte and Y. C. Eldar. Structured compressed sensing: From theory to applications. *IEEE Transactions on Signal Processing*, 59(9):4053–4085, 2011.

[73] N. G. Duffield and F. Lo Presti. Multicast inference of packet delay variance at interior network links. In *Proceedings of the IEEE Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies*, pages 1351–1360, 2000.

[74] B. Dumitrescu. *Positive trigonometric polynomials and signal processing applications*. Springer, 2007.

[75] M. A. Ebert. Possibilities for intensity-modulated brachytherapy: technical limitations on the use of non-isotropic sources. *Physics in medicine and biology*, 47(14):2495, 2002.

[76] M. A. Ebert. Potential dose-conformity advantages with multi-source intensity-modulated brachytherapy (IMBT). *Australasian Physics & Engineering Sciences in Medicine*, 29(2):165–171, 2006.

[77] Y. C. Eldar and G. Kutyniok. *Compressed sensing: theory and applications*. Cambridge University Press, 2012.

[78] J. Fang, D. Huiping, J. Li, H. Li, and R. S. Blumn. Super-resolution compressed sensing: A generalized iterative reweighted ℓ_2 approach. *arXiv preprint arXiv:1412.2477*, 2014.

[79] A. Fannjiang and W. Liao. Coherence pattern-guided compressive sensing with unresolved grids. *SIAM Journal on Imaging Sciences*, 5(1):179–202, 2012.

[80] A. C. Fannjiang. The MUSIC algorithm for sparse objects: a compressed sensing analysis. *Inverse Problems*, 27(3):35013–35044, 2011.

[81] L. Fejér. Über trigonometriche polynome. *Journal für die Reine und Angewandte Mathematik*, 146:53–82, 1915. in German.

[82] J. R. Fienup. Phase retrieval algorithms: a comparison. *Applied optics*, 21(15):2758–2769, 1982.

[83] M. H. Firooz and S. Roy. Network tomography via compressed sensing. In *Proceedings of IEEE Global Telecommunications Conference (GLOBECOM)*, pages 1–5, 2010.

[84] R. F. Fischer. *Precoding and signal shaping for digital transmission*. John Wiley & Sons, 2005.

[85] C. Fougner and S. Boyd. Parameter selection and pre-conditioning for a graph form solver. *arXiv preprint arXiv:1503.08366*, 2015.

[86] H. Gao. Robust fluence map optimization via alternating direction method of multipliers with empirical parameter optimization. *Physics in Medicine and Biology*, 61(7):2838, 2016.

[87] R. Ge, F. Huang, C. Jin, and Y. Yuan. Escaping from saddle points-online stochastic gradient for tensor decomposition.

[88] R. W. Gerchberg and W. O. Saxton. A practical algorithm for the determination of phase from image and diffraction plane pictures. *Optik*, 35(2):237–246, 1972.

[89] J. Ginsberg, M. H. Mohebbi, R. S. Patel, L. Brammer, M. S. Smolinski, and L. Brilliant. Detecting influenza epidemics using search engine query data. *Nature*, 457(7232):1012–1014, 2009.

[90] T. Goldstein, B. O'Donoghue, S. Setzer, and R. Baraniuk. Fast alternating direction optimization methods. *SIAM Journal on Imaging Sciences*, 7(3):1588–1623, 2014.

[91] M. Grant and S. P. Boyd. CVX: Matlab software for disciplined convex programming, version 2.0 beta. http://cvxr.com/cvx, September 2012.

[92] C. Haie-Meder, R. Pötter, E. Van Limbergen, E. Briot, M. De Brabandere, J. Dimopoulos, I. Dumas, T. P. Hellebust, C. Kirisits, S. Lang, et al. Recommendations from gynaecological (GYN) GEC-ESTRO working group (I): concepts and terms in 3D image based 3D treatment planning in cervix cancer brachytherapy with emphasis on MRI assessment of GTV and CTV. *Radiotherapy and Oncology*, 74(3):235–245, 2005.

[93] F. J. Harris. On the use of windows for harmonic analysis with the discrete Fourier transform. *Proceedings of the IEEE*, 66(1):51–83, 1978.

[94] B. He and X. Yuan. On the o(1/n) convergence rate of the douglas-rachford alternating direction method. *SIAM Journal on Numerical Analysis*, 50(2):700–709, 2012.

[95] C. Hegde, M. F. Duarte, and V. Cevher. Compressive sensing recovery of spike trains using a structured sparsity model. In *Proceedings of Signal Processing with Adaptive Sparse Structured Representations (SPARS)*, 2009.

[96] C.-J. Hsieh, K.-W. Chang, C.-J. Lin, S. S. Keerthi, and S. Sundararajan. A dual coordinate descent method for large-scale linear SVM. In *Proceedings of the 25th international conference on machine learning*, pages 408–415. ACM, 2008.

[97] C.-J. Hsieh, H.-F. Yu, and I. S. Dhillon. Passcode: Parallel asynchronous stochastic dual co-ordinate descent. In *Proceedings of the 32nd international conference on machine learning*, volume 15, 2015.

[98] Y. Hua. Esitmating two-dimensional frequencies by matrix enhancement and matrix pencil. *IEEE Transactions on Signal Processing*, 40(9):2267–2280, 1992.

[99] Y. Hua and T. K. Sarkar. Matrix pencil method for estimating parameters of exponentially dampled/undampled sinusoids in noise. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 38(5):814–824, 1990.

[100] B. Huang, M. Bates, and X. Zhuang. Super resolution fluorescence microscopy. *Annual review of biochemistry*, 78:993, 2009.

[101] T. Huang, Y. Liu, H. Meng, and X. Wang. Adaptive matching pursuit with constrained total least squares. *EURASIP Journal on Advances in Signal Processing*, 2012(1):1–12, 2012.

[102] Z. Huo and H. Huang. Distributed asynchronous dual free stochastic dual coordinate ascent. *arXiv preprint arXiv:1605.09066*, 2016.

[103] IBM. IBM ILOG CPLEX optimizer. https://www-01.ibm.com/software/commerce/optimization/cplex-optimizer/.

[104] D. Jacob, A. Raben, A. Sarkar, J. Grimm, and L. Simpson. Anatomy-based inverse planning simulated annealing optimization in high-dose-rate prostate brachytherapy: significant dosimetric advantage over other optimization techniques. *International Journal of Radiation Oncology, Biology, Physics*, 72(3):820–827, 2008.

[105] K. Jaganathan, J. Saunderson, M. Fazei, Y. C. Eldar, and B. Hassibi. Phaseless super-resolution using masks. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4039–4043, 2016.

[106] M. Jaggi, V. Smith, M. Takác, J. Terhorst, S. Krishnan, T. Hofmann, and M. I. Jordan. Communication-efficient distributed dual coordinate ascent. In *Advances in neural information processing systems*, pages 3068–3076, 2014.

[107] D. Jaruskova. Some problems with application of change-point detection methods to environmental data. *Environmetrics*, 8(5):469–483, 1997.

[108] S. Ji, Y. Xue, and L. Carin. Bayesian compressive sensing. *IEEE Transactions on Signal Processing*, 56(6):2346–2356, 2008.

[109] M. C. Joiner and A. Van der Kogel. *Basic clinical radiobiology*. CRC Press, 2016.

[110] A. Juditsky and A. Nemirovski. On verifiable sufficient conditions for sparse signal recovery via ℓ_1 minimization. *Mathematical programming*, 127(1):57–88, 2011.

[111] E. Karseras and W. Dai. Tracking dynamic sparse signals: a hierarchical kalman filter. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6546–6550, 2013.

[112] S. M. Kay. *Modern spectral estimation: theory and application*. Prentice Hall, 1988.

[113] R. H. Keshavan, A. Montanari, and S. Oh. Matrix completion from a few entries. *IEEE Transactions on Information Theory*, 56(6):2980–2998, 2010.

[114] M. A. Khajehnejad, W. Xu, A. S. Avestimehr, and B. Hassibi. Weighted ℓ_1 minimization for sparse recovery with prior information. In *Proceedings of IEEE International Symposium on Information Theory*, pages 483–487, 2009.

[115] C. Kirisits, S. Lang, J. Dimopoulos, D. Berger, D. Georg, and R. Pòtter. The vienna applicator for combined intracavitary and interstitial brachytherapy of cervical cancer: Design, application, treatment planning, and dosimetric results. *International Journal of Radiation Oncology, Biology, Physics*, 65(2):624 – 630, 2006.

[116] P. Kneschaurek, W. Schiessl, and R. Wehrmann. Volume-based dose optimization in brachytherapy. *International Journal of Radiation Oncology, Biology, Physics*, 45(3):811–815, 1999.

[117] S. L. Kukreja, J. Lofberg, and M. J. Brenner. A least absolute shrinkage and selection operator (lasso) for nonlinear system identification. In *Proceedings of International Federation of Automatic Control Symposium on System Identification*, pages 814–819, 2006.

[118] L. Lai, Y. Fan, and H. V. Poor. Quickest detection in cognitive radio: A sequential change detection framework. In *Proceedings of IEEE Global Telecommunications Conference (GLOBECOM)*, pages 1–5, 2008.

[119] L. Lai, H. V. Poor, Y. Xin, and G. Georgiadis. Quickest search over multiple sequences. *IEEE Transactions on Information Theory*, 57(8):5375–5386, 2011.

[120] K. Lee and Y. Bresler. Computing performance guarantees for compressed sensing. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5129–5132, 2008.

[121] E. Lessard and J. Pouliot. Inverse planning anatomy-based dose optimization for hdr-brachytherapy of the prostate using fast simulated annealing algorithm and dedicated objective function. *Medical physics*, 28(5):773–779, 2001.

[122] W. Liao and A. Fannjiang. MUSIC for single-snapshot spectral estimation: Stability and super-resolution. *arXiv preprint arXiv:1404.1484*, 2014.

[123] D. A. Linebarger, R. D. DeGroat, E. M. Dowling, P. Stoica, and G. L. Fudge. Incorporating a priori information into MUSIC-algorithms and analysis. *Signal Processing*, 46(1):85–104, 1995.

[124] X. Liu, C. Pelizzari, A. H. Belcher, Z. Grelewicz, and R. D. Wiersma. Use of proximal operator graph solver for radiation therapy inverse treatment planning. *Medical Physics*, 44(4):1246–1256, 2017.

[125] Y. Liu, R. T. Flynn, Y. Kim, and X. Wu. Asymmetric dose-volume optimization with smoothness control for rotating-shield brachytherapy. *Medical Physics*, 41(11):111709, 2014.

[126] Y. Liu, R. T. Flynn, W. Yang, Y. Kim, S. K. Bhatia, W. Sun, and X. Wu. Rapid emission angle selection for rotating-shield brachytherapy. *Medical physics*, 40(5):051720, 2013.

[127] F. Lo Presti, N. G. Duffield, J. Horowitz, and D. Towsley. Multicast-based inference of network-internal delay distributions. *IEEE/ACM Transactions on Networking*, 10(6):761–775, 2002.

[128] S. Ma, D. Goldfarb, and L. Chen. Fixed point and bregman iterative methods for matrix rank minimization. *Mathematical Programming*, 128(1-2):321–353, 2011.

[129] D. J. C. MacKay. *Information theory, inference and learning algorithms*. Cambridge University Press, 2003.

[130] M. Malloy and R. Nowak. On the limits of sequential testing in high dimensions. In *Proceedings of the Annual Asilomar Conference on Signals, Systems and Computers*, pages 1245–1249, 2011.

[131] M. Malloy and R. Nowak. Sequential analysis in high dimensional multiple testing and sparse recovery. In *Proceedings of IEEE International Symposium on Information Theory (ISIT)*, pages 2661–2665, 2011.

[132] M. Malloy, G. Tang, and R. Nowak. Quickest search for a rare distribution. In *Proceedings of the Annual Conference on Information Sciences and Systems (CISS)*, pages 1–6, 2012.

[133] A. Manikandan, B. Sarkar, V. T. Rajendran, P. R. King, N. V. M. Sresty, R. Holla, S. Kotur, and S. Nadendla. Role of step size and max dwell time in anatomy based inverse optimization for prostate implants. *Journal of medical physics/Association of Medical Physicists of India*, 38(3):148, 2013.

[134] S. L. Marple Jr. *Digital spectral analysis with applications*. Englewood Cliffs, NJ, Prentice-Hall, Inc., 1987.

[135] C. D. Meyer. *Matrix analysis and applied linear algebra*, volume 2. Siam, 2000.

[136] M. Mishali and Y. C. Eldar. From theory to practice: Sub-Nyquist sampling of sparse wideband analog signals. *IEEE Journal of Selected Topics in Signal Processing*, 4(2):375–391, 2010.

[137] K. V. Mishra, M. Cho, A. Kruger, and W. Xu. Super-resolution line spectrum estimation with block priors. In *Proceedings of Asilomar Conference on Signals, Systems and Computers*, pages 1211–1215, 2014.

[138] K.V. Mishra, M. Cho, A. Kruger, and W. Xu. Spectral super-resolution with prior knowledge. *IEEE Transactions on Signal Processing*, 63(20):5342–5357, 2015.

[139] K. Mohan and M. Fazel. Iterative reweighted algorithms for matrix rank minimization. *Journal of Machine Learning Research*, 13(1):3441–3473, 2012.

[140] A. Moitra. The threshold for super-resolution via extremal functions. *arXiv preprint arXiv:1408.1681*, 2014.

[141] MOSEK ApS. *The MOSEK optimization toolbox for MATLAB manual. Version 7.1 (Revision 31)*, 2015.

[142] D. Needell. Noisy signal recovery via iterative reweighted l1-minimization. In *Proceedings of Asilomar Conference on Signals, Systems, and Computers*, pages 113–117, 2009.

[143] Y. Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*. Kluwer, 2004.

[144] Y. Nesterov, H. Wolkowicz, and Y. Ye. *Semidefinite programming relaxations of nonconvex quadratic optimization*. Springer, 2000.

[145] F. Nielsen. Chernoff information of exponential families. *http://arxiv.org/abs/1102.2684*, 2011.

[146] V. F. Pisarenko, A. F. Kushnir, and I. V. Savin. Statistical adaptive algorithms for estimation of onset moments of seismic phases. *Physics of the Earth and Planetary Interiors*, 47:4–10, 1987.

[147] H. V. Poor and O. Hadjiliadis. *Quickest Detection*. Cambridge University Press, 2008.

[148] R. Pòtter, J. Dimopoulos, P. Georg, S. Lang, C. Waldhàusl, N. Wachter-Gerstner, H. Weitmann, A. Reinthaller, T. H. Knocke, S. Wachter, and C. Kirisits. Clinical impact of MRI assisted dose volume adaptation and dose escalation in brachytherapy of locally advanced cervix cancer. *Radiotherapy and Oncology*, 83(2):148 – 155, 2007.

[149] R. Pötter, C. Haie-Meder, E. Van Limbergen, I. Barillot, M. De Brabandere, J. Dimopoulos, I. Dumas, B. Erickson, S. Lang, A. Nulens, P. Petrow, J. Rownd, and C. Kirisits. Recommendations from gynaecological (GYN) GEC-ESTRO working group (II): Concepts and terms in 3D image-based treatment planning in cervix cancer brachytherapy-3D dose volume parameters and aspects of 3D image-based anatomy, radiation physics, radiobiology. *Radiotherapy and Oncology*, 78(1):67–77, 2006.

[150] R. Prony. Essai experimental et analytique. *J. de lÉcole Polytechnique (Paris)*, 1(2):24–76, 1795.

[151] B. Recht, M. Fazel, and P. A. Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM review*, 52(3):471–501, 2010.

[152] B. Recht, W. Xu, and B. Hassibi. Null space conditions and thresholds for rank minimization. *Mathematical programming*, 127(1):175–202, 2011.

[153] W. D. Renner, T. P. OConnor, and N. M. Bermudez. An algorithm for generation of implant plans for high-dose-rate irradiators. *Medical physics*, 17(1):35–40, 1990.

[154] M. J. Rivard, S. D. Davis, L. A. DeWerd, T. W. Rusch, and S. Axelrod. Calculated and measured brachytherapy dosimetry parameters in water for the Xoft Axxent X-Ray source: An electronic brachytherapy sourcea). *Medical physics*, 33(11):4020–4032, 2006.

[155] R. Roy and T. Kailath. Esprit-estimation of signal parameters via rotational invariance techniques. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 37(7):984–995, 1989.

[156] T. K. Sarkar and O. Pereira. Using the matrix pencil method to estimate the parameters of a sum of complex exponentials. *IEEE Antennas and Propagation Magazine*, 37(1):48–55, 1995.

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

[158] J. T. Schwartz. Probabilistic algorithms for verification of polynomial identities. In *Proceedings of International Symposiumon on Symbolic and Algebraic Computation*, pages 200–215, 1979.

[159] S. Shalev-Shwartz and T. Zhang. Stochastic dual coordinate ascent methods for regularized loss minimization. *Journal of Machine Learning Research*, 14:567–599, 2013.

[160] M. Sipser and D. Spielman. Expander codes. *IEEE transactions on Information Theory*, 42(6):1710–1722, 1996.

[161] M. I. Skolnik. *Radar handbook*. McGraw-Hill, third edition, 2008.

[162] S. Sra, S. Nowozin, and S. J. Wright. *Optimization for Machine Learning*. The MIT Press, 2011.

[163] P. Stoica and R. L. Moses. *Spectral analysis of signals*. Prentice Hall, 2005.

[164] M. Stojnic, F. Parvaresh, and B. Hassibi. On the reconstruction of block-sparse signals with an optimal number of measurements. *IEEE Transactions on Signal Processing*, 57(8):3075–3085, 2009.

[165] E. Stolyarova, K. T. Rim, S. Ryu, J. Maultzsch, P. Kim, L. E. Brus, T. F. Heinz, M. S. Hybertsen, and G. W. Flynn. High-resolution scanning tunneling microscopy imaging of mesoscopic graphene sheets on an insulating surface. *Proceedings of the National Academy of Sciences*, 104(22):9209–9212, 2007.

[166] Z. Tan and A. Nehorai. Sparse direction of arrival estimation using co-prime arrays with off-grid targets. *IEEE Signal Processing Letters*, 21(1):26–29, 2014.

[167] G. Tang, B. N. Bhaskar, and B. Recht. Sparse recovery over continuous dictionaries: Just discretize. In *Proceedings of Asilomar Conference on Signals, Systems, and Computers*, pages 1043–1047, 2013.

[168] G. Tang, B. N. Bhaskar, P. Shah, and B. Recht. Compressed sensing off the grid. *IEEE Transactions on Information Theory*, 59(11):7465–7490, 2013.

[169] G. Tang and A. Nehorai. Verifiable and computable ℓ_∞ performance evaluation of ℓ_1 sparse signal recovery. In *Proceedings of the Annual Conference on Information Sciences and Systems (CISS)*, pages 1–6, 2011.

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

[171] A. M. Tillmann and M. E. Pfetsch. The computational complexity of the restricted isometry property, the nullspace property, and related concepts in compressed sensing. *IEEE Transactions on Information Theory*, 60(2):1248–1259, 2014.

[172] O. Toeplitz. Zur theorie der quadratischen und bilinearen formen von unendlichvienlen veräanderlichen. *Mathenatische Annalen*, 70(3):351–376, 1911.

[173] D. H. Trivett and A. Z. Robinson. Modified Prony method approach to echo-reduction measurements. *The Journal of the Acoustical Society of America*, 70:1166–1175, 1981.

[174] J. Tropp and A. C. Gilbert. Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Transactions on Information Theory*, 53(12):4655–4666, 2007.

[175] Y. Tsang, M. Coates, and R. D. Nowak. Network delay tomography. *IEEE Transactions on Signal Processing*, 51(8):2125–2136, 2003.

[176] K. Tsianos, S. Lawlor, and M. G. Rabbat. Communication/computation tradeoffs in consensus-based distributed optimization. In *Advances in neural information processing systems*, pages 1943–1951, 2012.

[177] D. Tufts and R. Kumaresan. Estimation of frequencies of multiple sinusoids: Making linear prediction perform like maximum likelihood. In *Proceedings of the IEEE*, pages 975–989, 1982.

[178] R. H. Tütüncü, K. C. Toh, and M. J. Todd. Solving semidefinite-quadratic-linear programs using SDPT3. *Mathematical Programming*, 95(2):189–217, 2003.

[179] Y. Vardi. Network tomography: Estimating source-destination traffic intensities from link data. *Journal of the American Statistical Association*, 91(433):365–377, 1996.

[180] N. Vaswani. Kalman filtered compressed sensing. In *Proceedings of the 15th IEEE International Conference on Image Processing*, pages 893–896, 2008.

[181] N. Vaswani and W. Lu. Modified-CS: Modifying compressive sensing for problems with partially known support. *IEEE Transactions on Signal Processing*, 58(9):4595–4607, 2010.

[182] K. K. Velpula, A. A. Rehman, S. Chigurupati, R. Sanam, K. K. Inampudi, and C. S. Akila. Computational analysis of human and mouse CREB3L4 protein. *Bioinformation*, 8(12):574, 2012.

[183] M. J. Webster, D. J. Scanderbeg, W. T. Watkins, J. Stenstrom, J. D. Lawson, and W. Y. Song. Dynamic modulated brachytherapy (DMBT): Concept, design, and system development. *Brachytherapy*, 10:S33–S34, 2011.

[184] D. Wipf and S. Nagarajan. Iterative reweighted ℓ_1 and ℓ_2 methods for finding sparse solutions. *IEEE Journal of Selected Topics in Signal Processing*, 4(2):317–329, 2010.

[185] P. Wirfält, G. Bouleux, M. Jansson, and P. Stoica. Subspace-based frequency estimation utilizing prior information. In *Proceedings of IEEE Statistical Signal Processing Workshop*, pages 533–536, 2011.

[186] Y. Xia and D. Tse. Inference of link delay in communication networks. *IEEE Journal on Selected Areas in Communications*, 24(12):2235–2248, 2006.

[187] W. Xu, J.-F. Cai, K. V. Mishra, M. Cho, and A. Kruger. Precise semidefinite programming formulation of atomic norm minimization for recovering d-dimensional ($d \geq 2$) off-the-grid frequencies. In *Proceedings of IEEE Information Theory and Applications Workshop*, pages 1–4, 2014.

[188] W. Xu, J.-F. Cai, K. V. Mishra, M. Cho, and A. Kruger. Precise semidefinite programming formulation of atomic norm minimization for recovering d-dimensional ($d \geq 2$) off-the-grid frequencies. In *Proceedings of IEEE Information Theory and Applications Workshop (ITA)*, pages 1–4, 2014.

[189] W. Xu and B. Hassibi. Efficient compressive sensing with deterministic guarantees using expander graphs. In *Proceedings of IEEE Information Theory Workshop*, pages 414–419, 2007.

[190] W. Xu and B. Hassibi. Precise stability phase transitions for l1 minimization: A unified geometric framework. *IEEE Transactions on Information Theory*, 57(10):6894–6919, 2011.

[191] W. Xu, E. Mallada, and A. Tang. Compressive sensing over graphs. In *Proceedings of the 30th IEEE International Conference on Computer Communizations (INFOCOM)*, pages 2087–2095, 2011.

[192] T. Yang. Trading computation for communication: Distributed stochastic dual coordinate ascent. In *Advances in neural information processing systems*, pages 629–637, 2013.

[193] W. Yang, Y. Kim, X. Wu, Q. Song, Y. Liu, S. K. Bhatia, W. Sun, and R. T. Flynn. Rotating-shield brachytherapy for cervical cancer. *Physics in Medicine and Biology*, 58(11):3931–3941, 2013.

[194] Z. Yang and L. Xie. Enhancing sparsity and resolution via reweighted atomic norm minimization. *arXiv preprint arXiv:1408.5750*, 2014.

[195] G. Yu and G. Sapiro. Statistical compressive sensing of gaussian mixture models. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing(ICASSP)*, pages 3728–3731, 2011.

[196] L. Yu, H. Sun, J.-P. Barbot, and G. Zheng. Bayesian compressive sensing for cluster structured sparse signals. *Signal Processing*, 92(1):259–269, 2012.

[197] D. Zachariah, P. Wirfält, M. Jansson, and S. Chatterjee. Line spectrum estimation with probabilistic priors. *Signal Processing*, 93(11):2969–2974, 2013.

[198] M. Zarepisheh, L. Xing, and Y. Ye. A computation study on an integrated alternating direction method of multipliers for large scale optimization. *Optimization Letters*, pages 1–13, 2017.

[199] H. F. Zhang, K. Maslov, G. Stoica, and L. V. Wang. Functional photoacoustic microscopy for high-resolution and noninvasive in vivo imaging. *Nature biotechnology*, 24(7):848–851, 2006.

[200] S.-Y. Zhao and W.-J. Li. Fast asynchronous parallel stochastic gradient descent: A lock-free approach with convergence guarantee. In *Proceedings of AAAI Conference on Artificial Intelligence*, 2016.

[201] R. Zippel. Probabilistic algorithms for sparse polynomials. In *Proceedings of International Symposium on Symbolic and Algebraic Computation*, pages 216–226, 1979.

[202] J. Zygarlicki and J. Mroczka. Prony's method used for testing harmonics and interharmonics in electrical power systems. *Metrology and Measurement Systems*, 19(4):659–672, 2012.