Washington University in St. Louis

Washington University Open Scholarship

All Theses and Dissertations (ETDs)

1-1-2011

Computable Performance Analysis of Recovering Signals with Low-dimensional Structures

Gongguo Tang
Washington University in St. Louis

Follow this and additional works at: https://openscholarship.wustl.edu/etd

Recommended Citation

Tang, Gongguo, "Computable Performance Analysis of Recovering Signals with Low-dimensional Structures" (2011). *All Theses and Dissertations (ETDs)*. 647.
https://openscholarship.wustl.edu/etd/647

This Dissertation is brought to you for free and open access by Washington University Open Scholarship. It has been accepted for inclusion in All Theses and Dissertations (ETDs) by an authorized administrator of Washington University Open Scholarship. For more information, please contact digital@wumail.wustl.edu.

WASHINGTON UNIVERSITY IN ST. LOUIS

School of Engineering and Applied Science

Department of Electrical and Systems Engineering

Thesis Examination Committee:
Arye Nehorai, Chair
Martin Arthur
Renato Feres
Norman Katz
Jr-Shin Li
Mladen Victor Wickerhauser

Computable Performance Analysis of Recovering Signals with Low-dimensional Structures

by

Gongguo Tang

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

DOCTOR OF PHILOSOPHY

August 2011
Saint Louis, Missouri

copyright by
Gongguo Tang
2011

ABSTRACT OF THE THESIS

Computable Performance Analysis of Recovering Signals with Low-dimensional Structures

by

Gongguo Tang

Doctor of Philosophy in Electrical Engineering

Washington University in St. Louis, 2011

Research Advisor: Arye Nehorai

The last decade witnessed the burgeoning development in the reconstruction of signals by exploiting their low-dimensional structures, particularly, the sparsity, the block-sparsity, the low-rankness, and the low-dimensional manifold structures of general nonlinear data sets. The reconstruction performance of these signals relies heavily on the structure of the sensing matrix/operator. In many applications, there is a flexibility to select the optimal sensing matrix among a class of them. A prerequisite for optimal sensing matrix design is the computability of the performance for different recovery algorithms.

I present a computational framework for analyzing the recovery performance of signals with low-dimensional structures. I define a family of goodness measures for arbitrary sensing matrices as the optimal values of a set of optimization problems. As one of the primary contributions of this work, I associate the goodness measures with the fixed points of functions defined by a series of linear programs, second-order cone programs, or semidefinite programs, depending on the specific problem. This relation with the fixed-point theory, together with a bisection search implementation, yields efficient algorithms to compute the goodness measures with global convergence guarantees. As a by-product, we implement efficient algorithms to verify sufficient conditions for exact signal recovery in the noise-free case. The implementations perform orders-of-magnitude faster than the state-of-the-art techniques.

The utility of these goodness measures lies in their relation with the reconstruction performance. I derive bounds on the recovery errors of convex relaxation algorithms in terms of these goodness measures. Using tools from empirical processes and generic chaining, I analytically demonstrate that as long as the number of measurements are relatively large, these goodness measures are bounded away from zeros for a large class of random sensing matrices, a result parallel to the probabilistic analysis of the restricted isometry property. Numerical experiments show that, compared with the restricted isometry based performance bounds, our error bounds apply to a wider range of problems and are tighter, when the sparsity levels of the signals are relatively low. I expect that computable performance bounds would open doors for wide applications in compressive sensing, sensor arrays, radar, MRI, image processing, computer vision, collaborative filtering, control, and many other areas where low-dimensional signal structures arise naturally.

Acknowledgments

I am grateful to my advisor, Professor Nehorai, for his wonderful mentoring, inspiring ideas, and continuous support in both life and research. He has been providing me with a stimulating environment to pursue the topics I like most. I wish to thank my defense committee members, Prof. Martin Arthur, Prof. Norman Katz, Prof. Jr-Shin Li, Prof. Renato Feres, and Prof. Mladen Victor Wickerhauser, for their valuable suggestions and comments.

I am indebted to my friends and lab mates. I would especially like to thank Patricio, Pinaki, and Murat for their patient help, Satya, Vanessa, Marija, and Peng for collaboration on exciting topics, Sandeep, Phani, Kofi, Xiaoxiao, and Elad for stimulating discussions, as well as Tao and Weifeng for the fun times.

I must thank my father and mother for their constant support at both good and bad times. I am also indebted to my brother who inspired me to explore the beauty of science, engineering, and mathematics. Most importantly, I am deeply grateful to my wonderful wife Xiaomao for her unlimited and unconditional love, support, and encouragement, without which I would still be stuck in my first paper.

A special thanks goes to Mr. James Ballard and Ms. Lynnea Brumbaugh at the Engineering Communication Center, Washington University in St. Louis, who sat with me for hours to polish the English of my papers many times.

Gongguo Tang

Washington University in Saint Louis
August 2011

Dedicated to Xiaomao.

Contents

Abstract . . . . **ii**

Acknowledgments . . . . **iv**

List of Tables . . . . **ix**

List of Figures . . . . **x**

1 Introduction **1**
- 1.1 Applications 3
- 1.2 Need for Computable Performance Analysis 8
- 1.3 Contributions 11
- 1.4 Organization 11

2 Mathematical Foundations **12**
- 2.1 Vector and Matrix 12
  - 2.1.1 Basic Notations 12
  - 2.1.2 Norms 14
  - 2.1.3 Inequalities 16
- 2.2 Optimization 18
  - 2.2.1 Introduction 18
  - 2.2.2 Berge's Maximum Theorem 19
  - 2.2.3 Lagrange Duality 19
  - 2.2.4 Karush-Kuhn-Tucker (KKT) conditions 20
  - 2.2.5 Linear, Second-Order Cone, and Semidefinite Programs 21
  - 2.2.6 Subdifferential and Nonsmooth Optimization 23
  - 2.2.7 Fixed Point Iteration 25
- 2.3 Probability 26
  - 2.3.1 Notations 26
  - 2.3.2 Random Sensing Ensembles 27
  - 2.3.3 Gaussian Process 27
  - 2.3.4 Estimates for Empirical Processes 31

3 Sparsity Recovery: Background **35**
- 3.1 Introduction to Sparsity Recovery 35
- 3.2 Recovery Algorithms 37
- 3.3 Null Space Property and Restricted Isometry Property 40

3.4 Probabilistic Analysis . . . 42

4 Computable Performance Analysis for Sparsity Recovery . . . 43
4.1 Goodness Measures and Error Bounds . . . 43
4.2 Verification Algorithm . . . 48
4.3 Computation Algorithm . . . 49
4.4 Probabilistic Analysis . . . 61
4.5 Numerical Simulations . . . 66

5 Block Sparsity Recovery: Background . . . 81
5.1 Introduction to Block Sparsity Recovery . . . 81
5.2 Recovery Algorithms . . . 82
5.3 Null Space Characterization and Restricted Isometry Property . . . 83
5.4 Probabilistic Analysis . . . 85

6 Computable Performance Analysis for Block-Sparsity Recovery . . . 86
6.1 Goodness Measures and Error Bounds . . . 86
6.2 Verification Algorithm . . . 92
    6.2.1 Semidefinite Relaxation for Verification . . . 92
    6.2.2 Smoothing Technique for Solving (6.33) . . . 95
6.3 Computation Algorithm . . . 101
    6.3.1 Fixed-Point Iteration for Computing $\omega_\diamond(\cdot)$ . . . 101
    6.3.2 Relaxation of the Subproblem . . . 108
    6.3.3 Smoothing Technique for Solving (6.86) . . . 110
    6.3.4 Smoothing Technique for Solving (6.88) . . . 112
    6.3.5 Fixed-Point Iteration for Computing a Lower Bound on $\omega_\diamond$ . . . 113
6.4 Probabilistic Analysis . . . 120
6.5 Preliminary Numerical Simulations . . . 123

7 Low-Rank Matrix Recovery: Background . . . 127
7.1 Introduction to Low-Rank Matrix Recovery . . . 127
7.2 Recovery Algorithms . . . 128
7.3 Null Space Characterization and Restricted Isometry Property . . . 130
7.4 Probabilistic Analysis . . . 132

8 Partial Extension to Low-Rank Matrix Recovery . . . 133
8.1 ℓ_*-Constrained Minimal Singular Values . . . 133
    8.1.1 Stability of Convex Relaxation Algorithms . . . 135
    8.1.2 Basis Pursuit . . . 137
    8.1.3 Dantzig Selector . . . 139
    8.1.4 LASSO Estimator . . . 141
8.2 Probabilistic Analysis . . . 143
8.3 Computational Difficulties . . . 145

9 Conclusions and Future Work **147**
9.1 Conclusions 147
9.2 Future Work 148

References **151**

Vita **159**

List of Tables

4.1 Comparison of L_∞ and JN for a Hadamard matrix with leading dimension $n = 256$. 68

4.2 Comparison of L_∞ and JN for a Gaussian matrix with leading dimension $n = 256$. 69

4.3 Comparison of L_∞ and JN for Gaussian and Hadamard matrices with leading dimension $n = 1024$. In the column head, "G" represents Gaussian matrix and "H" represents Hadamard matrix. 69

4.4 Comparison of the ω_2 based bounds and the RIC based bounds on the ℓ_2 norms of the errors of the Basis Pursuit algorithm for a Bernoulli matrix with leading dimension $n = 256$. 71

4.5 Comparison of the ω_2 based bounds and the RIC based bounds on the ℓ_2 norms of the errors of the Basis Pursuit algorithm for a Hadamard matrix with leading dimension $n = 256$. 72

4.6 Comparison of the ω_2 based bounds and the RIC based bounds on the ℓ_2 norms of the errors of the Basis Pursuit algorithm for a Gaussian matrix with leading dimension $n = 256$. 73

4.7 Comparison of the ω_∞ based bounds and the RIC based bounds on the ℓ_2 norms of the errors of the Dantzig selector algorithm for the Bernoulli matrix used in Table 4.4. 74

4.8 Comparison of the ω_∞ based bounds and the RIC based bounds on the ℓ_2 norms of the errors of the Dantzig selector algorithm for the Hadamard matrix used in Table 4.5. 75

4.9 Comparison of the ω_∞ based bounds and the RIC based bounds on the ℓ_2 norms of the errors of the Dantzig selector algorithm for the Gaussian matrix used in Table 4.6. 76

4.10 Time in seconds taken to compute $\omega_2(A, \cdot)$ and $\omega_\infty(A^T A, \cdot)$ for Bernoulli, Hadamard, and Gaussian matrices 77

6.1 Comparison of the sparsity level bounds on the block-sparse model and the sparse model for a Gaussian matrix $A \in \mathbb{R}^{m \times np}$ with $n = 4, p = 60$. 124

6.2 $\omega_2(A, 2k)$ and $\delta_{2k}(A)$ computed for a Gaussian matrix $A \in \mathbb{R}^{m \times np}$ with $n = 4$ and $p = 60$. 126

6.3 The $\omega_2(A, 2k)$ based bounds on the ℓ_2 norms of the errors of the BS-BP for the Gaussian Matrix in Table 6.2. 126

List of Figures

1.1 Signals with low-dimensional structures. Top to bottom: sparse signal, block-sparse signal, and low-rank matrix.) 2

1.2 Signal acquisition paradigm of compressive sensing. (Modified from Richard Baraniuk, Justin Romberg, and Michael Wakin's slides "Tutorial on Compressive Sensing".) 3

1.3 Far-field electromagnetic imaging. 5

1.4 Exploiting sparsity (in the Heaviside domain) improves MRI recovery. (Modified from [1].) 6

1.5 Single image superresolution results of the girl image magnified by a factor of 3 and the corresponding RMSEs. Left to right: input, bicubic interpolation (RMSE: 6.843), neighborhood embedding [2] (RMSE: 7.740), sparse representation [3] (RMSE: 6.525), and the original. (Reproduced from [3].) 6

1.6 Sparse representation for robust face recognition. (a) Test image with occlusion. (b) Test image with corruption. (Reproduced from [4].) . 7

1.7 Collaborative filtering as a matrix completion problem. 8

1.8 The sampling trajectory affects the system performance. 9

2.1 Random sensing matrices. Top to bottom: Gaussian matrix, Bernoulli matrix, real part of a Fourier matrix. 28

4.1 The functions $f_s(\eta)$ and $f_{s,i}(\eta)$. The blue line is for the diagonal $f = \eta$. The thick black plot is for $f_s(\eta)$, and other colored thin plots are for $f_{s,i}(\eta)$s. 52

4.2 Illustration of the proof for $\rho > 1$. 55

4.3 Illustration of the proof for $f_s(\eta^*) \geq \rho\eta^*$. 57

4.4 Illustration of naive fixed-point iteration (4.54) when $\diamond = \infty$. 59

4.5 $\omega_2(A, 2k)$ and $\omega_\infty(A^TA, 2k)$ as a function of k with $n = 512$ and different ms. The top plot is for a Gaussian matrix, and the bottom plot is for a Hadamard matrix. 78

4.6 $\omega_2(A, 2k)$ based bounds v.s. RIC based bounds on the ℓ_2 norms of the errors for a Gaussian matrix with leading dimension $n = 512$. Left: $\omega_2(A, 2k)$ based bounds; Right: RIC based bounds. 79

4.7 $\omega_\infty(A^TA, 2k)$ based bounds v.s. RIC based bounds on the ℓ_2 norms of the errors for a Hadamard matrix with leading dimension $n = 512$. Left: $\omega_2(A, 2k)$ based bounds; Right: RIC based bounds 80

6.1 Illustration of the proof for $\rho > 1$. 105
6.2 Illustration of the proof for $f_s(\eta^*) \geq \rho\eta^*$. 106

Chapter 1

Introduction

Recovery of signals with low-dimensional structures, in particular, sparsity [5], block-sparsity [6], low-rankness [7], and and low-dimensional manifold structures for general non-linear data set [8,9] has found numerous applications in signal sampling, control, inverse imaging, remote sensing, radar, sensor arrays, image processing, computer vision, and so on. Mathematically, the recovery of signals with low-dimensional structures aims to reconstruct a signal with a prescribed structure, from usually noisy linear measurements:

$$\boldsymbol{y} = A\boldsymbol{x} + \boldsymbol{w}, \tag{1.1}$$

where $\boldsymbol{x} \in \mathbb{R}^N$ is the signal to be reconstructed, $\boldsymbol{y} \in \mathbb{R}^m$ is the measurement vector, $A \in \mathbb{R}^{m \times N}$ is the sensing/measurement matrix, and $\boldsymbol{w} \in \mathbb{R}^m$ is the noise. For example, a sparse signal is assumed to have only a few non-zero coefficients when represented as a linear combination of atoms from an orthogonal basis or from an overcomplete dictionary. For a block-sparse signal, the non-zero coefficients are assumed to cluster into blocks; low-rank matrices have sparse singular value vectors. The signals with sparsity, block-sparsity, and low-rank structures are illustrated in Figure 1.1. Exploiting these low-dimensional structures can greatly improve the recovery performance even when only a small number of measurements are available.

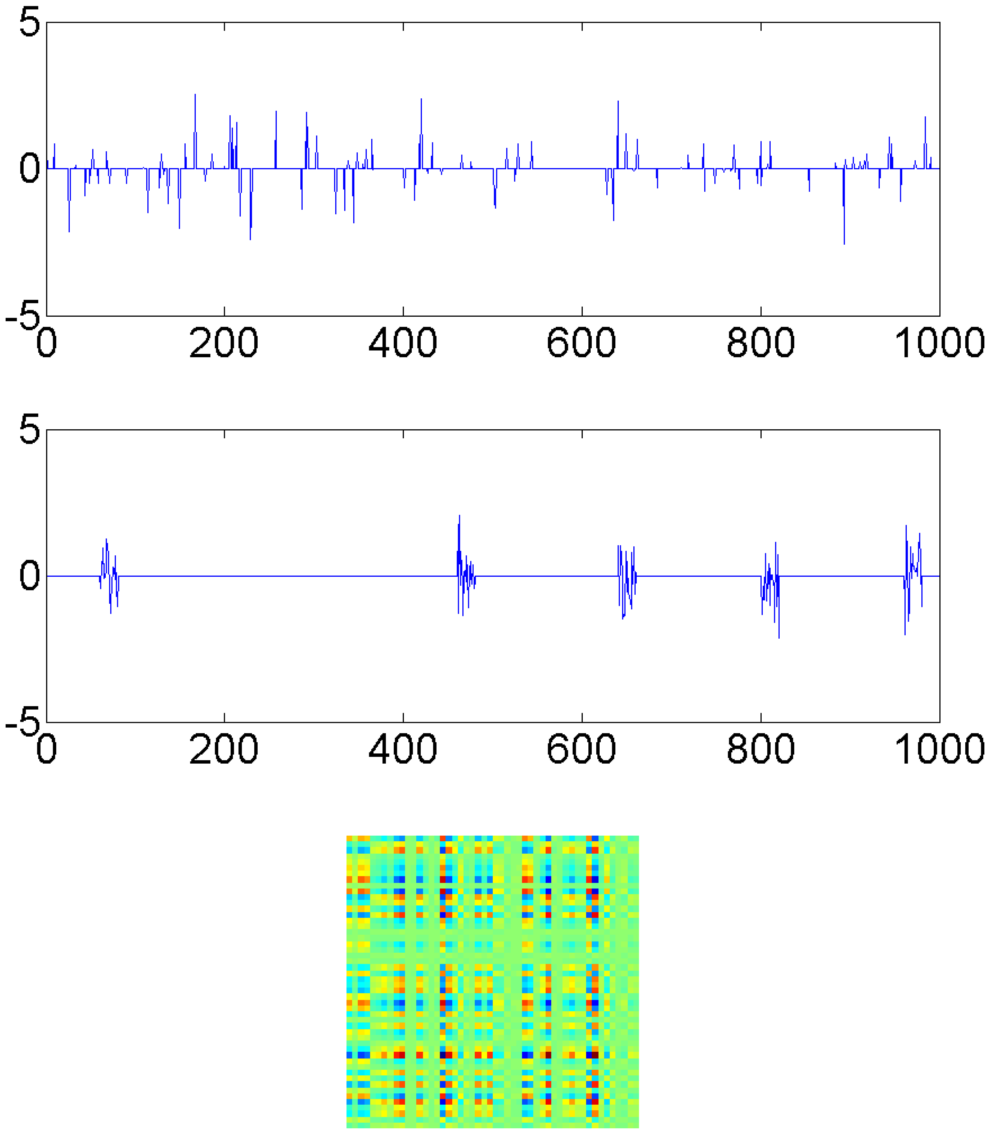


Figure 1.1: Signals with low-dimensional structures. Top to bottom: sparse signal, block-sparse signal, and low-rank matrix.)

1.1 Applications

We list some applications of the recovery of low-dimensional signals. This list is by no means exhaustive. The readers are encouraged to check the references therein for more detailed account on these applications and many other related ones.

Compressive Sensing

Compressive sensing [5,10,11] is a new paradigm for signal acquisition. The investigation of compressive sensing renewed the research community's interest in the recovery of signals with low-dimensional structures and brought it to great popularity. The traditional sampling-and-compression scheme for signal acquisition is extremely wasteful. As exemplified in [5], "*one can think about a digital camera which has millions of imaging sensors, the pixels, but eventually encodes the picture in just a few hundred kilobytes.*" Compressive sensing properly combines the sampling and compression processes by measuring the signal from different directions, *i.e.*, instead of sampling the signal, compressive sensing observes projections of the signal onto random directions. One surprising discovery revealed by the theory of compressive sensing is that linear, random, and non-adaptive measurements of a number essentially proportional to the degree of freedom of the signal is sufficient for exactly recovering the signal in the noise-free case, and stably reconstructing the signal if noise exists [5]. This new signal acquisition scheme greatly reduces the data necessary to record or transmit a signal by exploiting the sparsity structure for most natural signals. It pushes the information sampling rate far beyond the limit set by Shannon and Nyquist.

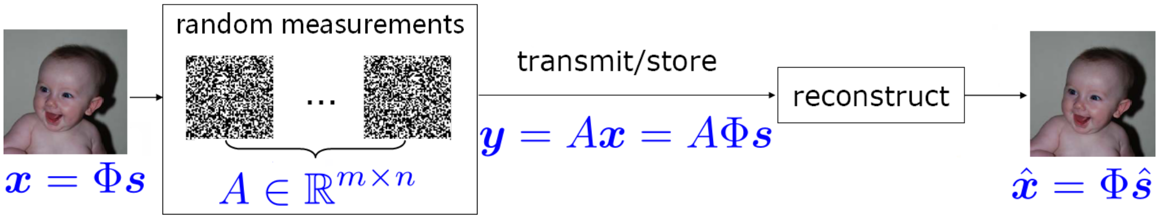

random measurements
...
$A \in \mathbb{R}^{m\times n}$
transmit/store
reconstruct
$\boldsymbol{x} = \Phi\boldsymbol{s}$
$\boldsymbol{y} = A\boldsymbol{x} = A\Phi\boldsymbol{s}$
$\hat{\boldsymbol{x}} = \Phi\hat{\boldsymbol{s}}$

Figure 1.2: Signal acquisition paradigm of compressive sensing. (Modified from Richard Baraniuk, Justin Romberg, and Michael Wakin's slides "Tutorial on Compressive Sensing".)

Sensor Arrays and Radar

Many practical problems in radar and sensor arrays can be reformulated as one of sparse spectrum estimation through parameter discretization. In [12], the authors transform the process of source localization using sensory arrays into the task of estimating the spectrum of a sparse signal by discretizing the parameter manifold. This method exhibits super-resolution in the estimation of direction of arrival (DOA) compared with traditional techniques such as beamforming [13], Capon [14], and MUSIC [15, 16]. Since the basic model employed in [12] applies to several other important problems in signal processing (see [17] and references therein), the principle is readily applicable to those cases. This idea is later generalized and extended to other source localization settings in [18–20].

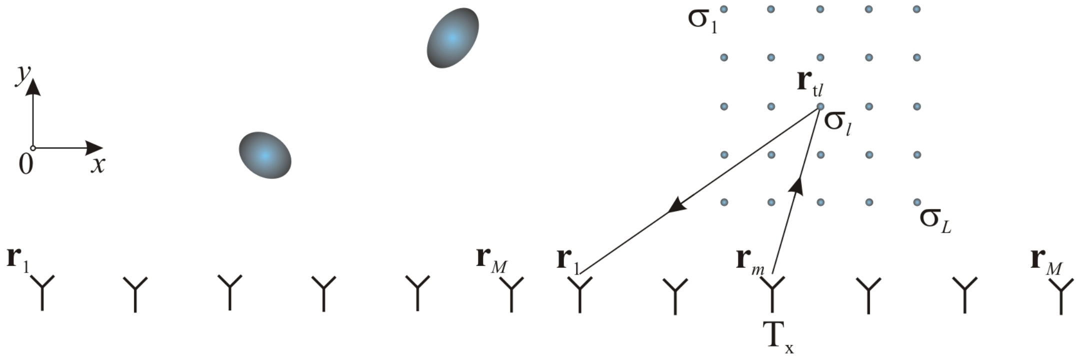
In radar applications, the target position-velocity space is usually discretized into a large set of grid points. Assuming the target parameters are within the grid point set, we represent the measurements as a linear combination of basis functions determined by the grid points and the radar system. Only those basis functions whose corresponding grid points have targets will have non-zero coefficients. Since there are only a few targets compared with the number of grid points, the coefficient vector is sparse or block-sparse. Therefore, the target estimation problem is transformed into one of estimating the sparse coefficients. This procedure generally produces estimates with super resolution [21, 22].

Electromagnetic Imaging

In electromagnetic imaging with a far-field assumption, the observed signals are modeled as attenuated and delayed replicas of the transmitted signal, with attenuation coefficients specified by the targets' scattering properties. Since the targets are sparse in the spatial domain, a discretization procedure transforms the model into one of sparse representation as shown in 1.3 and the targets could be accurately localized using sparsity enforcing techniques. A similar formulation is proposed for the near-field scenarios by employing the electric field integral equation to capture the mutual interference among targets. In both cases, exploiting sparsity of the targets in the spatial domain improves localization accuracy [23].

MRI

In magnetic resonance imaging (MRI), the images to be observed are usually approximately piece-wise constant, making them sparse in the Heaviside transform domain. MRI observes a subset of the Fourier coefficients of the underlying image. In Figure



(a) True model with unknown targets.

(b) Sparse model through spatial discretization.

Figure 1.3: Far-field electromagnetic imaging.

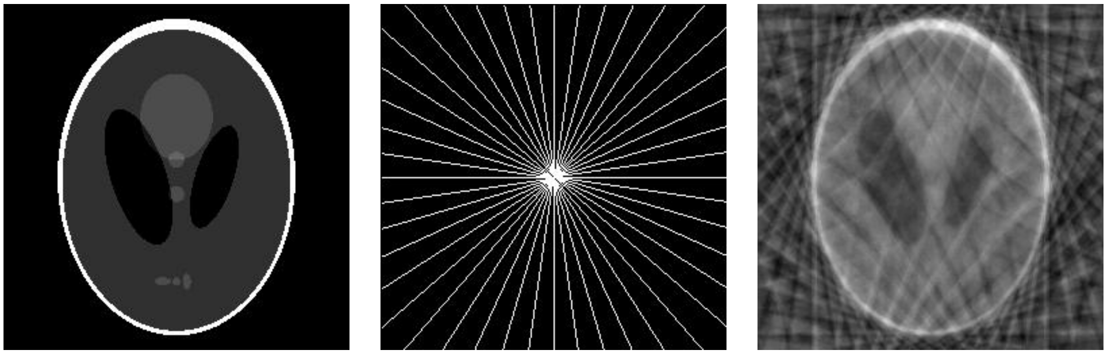
1.4 (b), Fourier coefficients are sampled along 22 approximately radial lines. Traditional techniques using minimal energy reconstruction, *i.e.*, minimizing the ℓ_2 norm subject to observation constraint, produced recovered images with a lot of artifacts, as shown in Figure 1.4 (c). In contrast, reconstruction by minimizing the ℓ_1 norm of the Heaviside transform, or the closely related total variation, both of which enforce sparsity in the transform domain, would exactly recover the original image as illustrated in Figure 1.4 (f) [1].

Image Processing

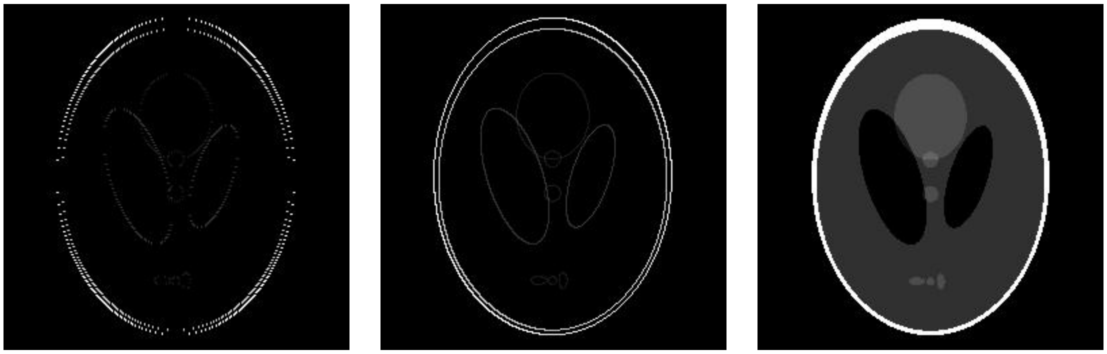
Most natural images and image patches are sparse in the wavelet [24] or discrete cosine [25] transform domains, learned dictionaries [26], or even randomly sampled raw image patches [3]. Exploiting this sparsity facilitates image denoising [27], image inpainting [28], image superresolution [3], and many other image processing tasks. For example, in [3], the authors propose a single image superresolution approach by finding the coefficients in the sparse representation of each patch of the low-resolution input, and using these coefficients to generate the high-resolution output. The results are compared with other techniques in Figure 1.5.

Computer Vision and Pattern Recognition

Sparse signal representation and reconstruction have seen a significant impact in computer vision, even when the task is to extract high-level sematic information [29]. In [4], a test face image with possible occlusion is represented as a sparse linear



(a) Logan-Shepp phantom (b) Fourier Sampling (c) Min-energy recovery



(d) Heaviside coefficients (e) Total variation (TV) coefficients (f) Min-TV recovery

Figure 1.4: Exploiting sparsity (in the Heaviside domain) improves MRI recovery. (Modified from [1].)

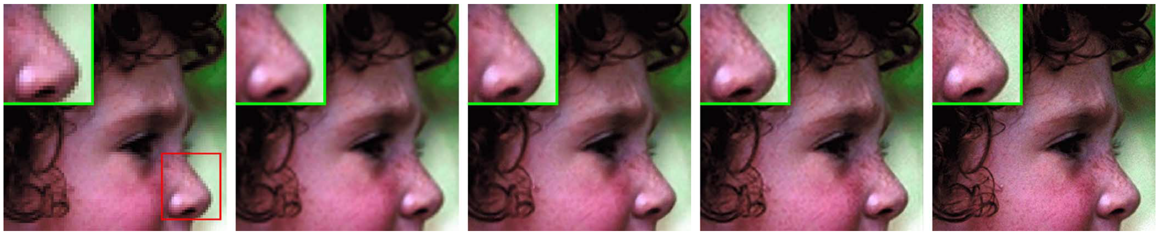


Figure 1.5: Single image superresolution results of the girl image magnified by a factor of 3 and the corresponding RMSEs. Left to right: input, bicubic interpolation (RMSE: 6.843), neighborhood embedding [2] (RMSE: 7.740), sparse representation [3] (RMSE: 6.525), and the original. (Reproduced from [3].)

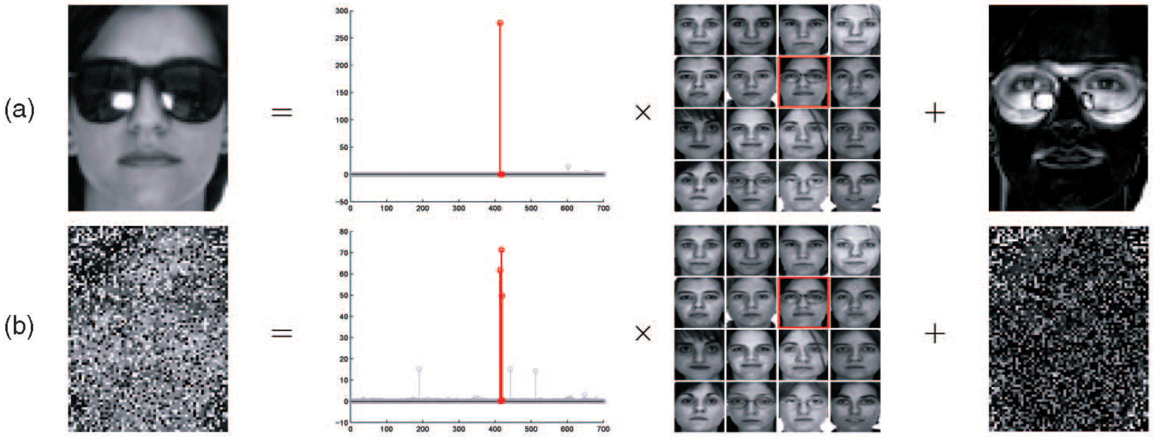
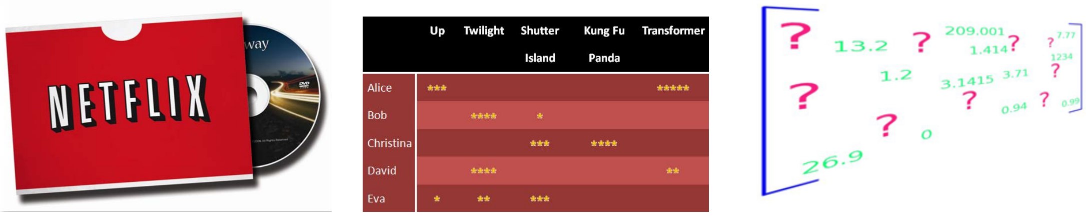


Figure 1.6: Sparse representation for robust face recognition. (a) Test image with occlusion. (b) Test image with corruption. (Reproduced from [4].)

combination of training samples plus sparse errors due to occlusion or corruption, as illustrated in Figure 1.6. The recovered spare coefficients simultaneously separate the occlusion from the face image and determines the identity of the test image, allowing robust face recognition.

Collaborative Filtering

Collaborative filtering is the process of automatically predicting a particular user's preference on something by collecting similar information from many other users. It is best illustrated by the Netflix prize open competition. As an online DVD-rental service, Netflix makes significant profit by accurately recommending DVDs to users according to their tastes. The training data set provided by Netflix for the competition consists of $100,480,507$ ratings given by $480,189$ users to $17,770$ movies. Mathematically, we are given a data matrix with $480,180$ rows (users), $17,770$ columns (movies), and $\frac{100{,}480{,}507}{480{,}189\times 17{,}770} \approx 1.18\%$ of its entries (see Figure 1.7 (b)). The task is to predict the missing entries from the available ones (see Figure 1.7 (c)). Apparently, without imposing additional structure on the data matrix, we are not able to make any inference on the missing entries as they could be any allowed numbers. A sensible assumption about the data matrix is that, due to the correlations between different users and different movies, its rows and columns are highly dependent, *i.e.*, it has a small rank. The problem of completing the missing entries of a low-rank matrix is called matrix completion [30–32], a special case of low-rank matrix recovery.



(a) Netflix prize (b) Data matrix (c) Matrix completion

Figure 1.7: Collaborative filtering as a matrix completion problem.

Control

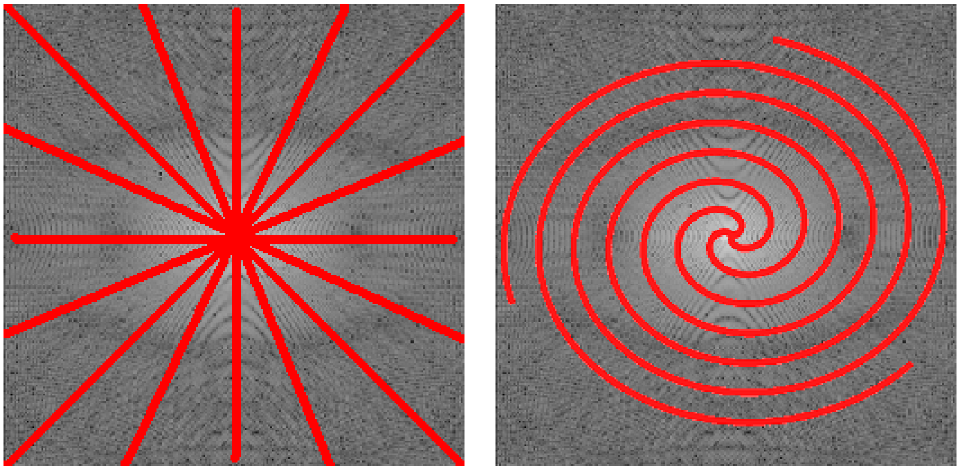
Low-rank matrices arise naturally in a variety of applications in control and system theory such as model reduction, minimum order control synthesis, and system identification [33]. Consider the system identification problem, the task is to decide the dimension of the system state vector and system matrices A, B, C, D from the input-output $\{\boldsymbol{u}(t), \boldsymbol{y}(t)\}_{t=1}^{T}$ according to the following model:

$$\begin{aligned} \boldsymbol{x}(t+1) &= A\boldsymbol{x}(t) + B\boldsymbol{u}(t), \\ \boldsymbol{y}(t) &= C\boldsymbol{x}(t) + D\boldsymbol{u}(t). \end{aligned} \tag{1.2}$$

This problem can be formulated as a low-rank matrix estimation problem. Refer to [33] for more details.

1.2 Need for Computable Performance Analysis

A theoretically justified way to exploit the low-dimensional structure in recovering $\boldsymbol{x}$ is to minimize a convex function that is known to enforce that low-dimensional structure. Examples include using the ℓ_1 norm to enforce sparsity, block-ℓ_1 norm (or ℓ_2/ℓ_1 norm) to enforce the block-sparsity, and the nuclear norm to enforce the low-rankness. The ability of these convex relaxation algorithms in recovering a low-dimensional signal in the noise-free case is guaranteed by various null space properties [34–38]. In the noisy case, the performance of these convex enforcements is usually analyzed using variants of the restricted isometry property (RIP) [6, 7, 39]. Upper bounds on the ℓ_2 norm of the error vectors for various recovery algorithms have been expressed in terms of the RIP. Unfortunately, it is extremely difficult to verify that the RIP of a



(a) Radial sampling trajectory (b) Spiral sampling trajectory

Figure 1.8: The sampling trajectory affects the system performance.

specific sensing matrix satisfies the conditions for the bounds to be valid, and even more difficult to directly compute the RIP itself. Actually, the only known sensing matrices with nice RIPs are certain types of random matrices [40].

In this dissertation, we investigate the recovery performance for sparse signals, block-sparse signals, and low-rank matrices. The aforementioned applications require a computable performance analysis of sparsity recovery, block-sparsity recovery, and low-rank matrix reconstruction. While it is perfectly reasonable to use a random matrix for signal sampling, the sensing matrices in other applications are far from random and actually depend on the underlying measurement devices and the physical processes that generate the observations. Due to the computational challenges associated with the RIP, it is necessary to seek computationally more amenable goodness measures of the sensing matrices. Computable performance measures would open the door to wide applications. Firstly, it provides a means to pre-determine the performance of the sensing system before its implementation and the taking of measurements. In addition, in radar imaging [22], sensor arrays [12], DNA microarrays [41], and MRI [42], we usually have the freedom to optimally design the sensing matrix. For example, in MRI the sensing matrix is determined by the sampling trajectory in the Fourier domain as shown in Figure 1.8; in radar systems the optimal sensing matrix design is connected with optimal waveform design, one of the major topics of radar signal processing. In view of the model (1.1), to optimally design the sensing matrix, we need to

1. analyze how the performance of recovering $\boldsymbol{x}$ from $\boldsymbol{y}$ is affected by A using performance bounds on reconstruction errors, and define a function $\omega(A)$ to accurately quantify the goodness of A in the context of signal reconstruction;
2. develop algorithms to efficiently verify that $\omega(A)$ satisfies the conditions for the bounds to hold, as well as to efficiently compute $\omega(A)$ for arbitrarily given A;
3. design mechanisms to select within a matrix class the sensing matrix that is optimal in the sense of best $\omega(A)$.

By proposing a family of goodness measures and designing algorithms with guaranteed global convergence, we will successfully address the first two points in this dissertation. We also point out ways to address the optimal sensing matrix design problem in future.

One distinctive feature of this work is our use of the ℓ_∞ and the block-ℓ_∞ norm as performance criteria for sparsity recovery and block-sparsity recovery, respectively. Although the ℓ_2 norm has been used as the performance criterion by the majority of published research in block-sparse signal recovery, the adoption of the ℓ_∞ norm and the block-ℓ_∞ norm is better justified. The ℓ_2 norms of the error vectors can be expressed in terms of the ℓ_∞ norm and the block-ℓ_∞ norm in a tight and non-trivial manner. More importantly, the ℓ_∞ norm and the block-ℓ_∞ norm of the error vector have direct connections with the support recovery problem. In practical applications, the signal support is usually physically more significant than the component values. For example, in radar imaging, the sparsity constraints are usually imposed on the discretized time–frequency domain. The distance and velocity of a target have a direct correspondence to the support of the recovered signal. The magnitude determined by coefficients of reflection is of less physical significance [21, 22]. Last but not least, the ℓ_∞ norm and the block-ℓ_∞ norm result in performance measures that are easier to compute.

1.3 Contributions

We preview our contributions. We first define a family of goodness measures of the sensing matrix, and use them to derive performance bounds on the recovery error vectors. Our preliminary numerical results show that these bounds are tighter than the existing restricted isometry constant based bounds. Secondly and most importantly, we develop a fixed point iteration framework to design algorithms that efficiently compute or bound the goodness measures for arbitrary sensing matrices. Each fixed point iteration solves a series of linear programs, second-order cone programs, or semidefinite programs, depending on the problem. The fixed point iteration framework also demonstrates the algorithms' convergence to the global optima from any initial point. As a by-product, we obtain a fast algorithm to verify the sufficient condition guaranteeing exact signal recovery via convex relaxation. Thirdly, we show that the goodness measures are non-degenerate for subgaussian and isotropic random sensing matrices as long as the number of measurements is relatively large, a result parallel to that of restricted isometry constant for random matrices.

1.4 Organization

The rest of the dissertation is organized as follows. In Chapter 2, we introduce our notations and present mathematical tools in matrix analysis, optimization, and probability and statistics that are necessary for the development of this work. In Chapter 3, we review previous work and present background knowledge in sparsity recovery. Chapter 4 is devoted to our computable performance analysis for sparsity recovery. We turn to block-sparsity recovery from Chapter 5 and summarize relevant background information. We then develop our computable performance analysis framework for block-sparsity recovery in Chapter 6. In Chapter 7, we introduce the problem of low-rank matrix recovery. We partially extend our results on sparsity recovery and block-sparsity recovery to low-rank matrix recovery in Chapter 8. Section 9 summarizes our conclusions and points out potential future work.

Chapter 2

Mathematical Foundations

In this chapter, we present the mathematical foundations of the dissertation by introducing notations and including basic concepts and relevant facts in matrix analysis, optimization, and probability theory.

2.1 Vector and Matrix

2.1.1 Basic Notations

Sets are denoted by either upper case letters such as S or scripted upper case letters such as $\mathcal{I}, \mathcal{F}, \mathcal{H}$. The symbols $\in$, $\subseteq$, $\subset$, $\bigcap$, $\bigcup$, and $\setminus$ denote the membership, subset, proper subset, intersection, union, and relative complement relations or operations, respectively. Specials sets such as the empty set, the set of all natural numbers, the set of all integers, the set of all real numbers, and the set of all complex numbers are represented by $\emptyset$, $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{R}$, and $\mathbb{C}$, respectively. The set of non-negative real numbers are denoted by $\mathbb{R}_+$. We use $\mathbb{F}$ to denote a general field. The cardinality of a set S, or the number of its elements, is expressed as $|S|$, either a non-negative integer or ∞.

The n-dimensional real (resp. complex) Euclidean space is $\mathbb{R}^n$ (resp. $\mathbb{C}^n$.) The n-dimensional Euclidean space over a general field $\mathbb{F}$ is denoted by $\mathbb{F}^n$. $\mathbb{R}_+^n$ has a similar meaning. We use bold, lower case letters such as $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}$ and $\boldsymbol{\xi}, \boldsymbol{\zeta}$ to represent generic column vectors in the Euclidean space $\mathbb{F}^n$. Non-bold lower case letters are for scalars. The ith component of a vector $\boldsymbol{x} \in \mathbb{F}^n$, as a scalar, is denoted by x_i, and $\boldsymbol{x}_\alpha$ is reserved for vectors parameterized by the subscript $\alpha \in \mathcal{I}$. However, suppose $S \subset \{1, \dots, n\}$

is an index set, we denote by $\boldsymbol{x}_S$ the vector in $\mathbb{F}^{|S|}$ formed by the components of $\boldsymbol{x}$ indexed by the set S. Whether the subscript is a parameter or a component index can be inferred from whether the letter is bold or not and from the context.

The symbols $\mathbf{0}$, $\mathbf{e}_i$, and $\mathbf{1}$ are reserved for the zero vector, the ith canonical basis vector, and the vector with all ones, respectively. The dimensions of these vectors, if not explicitly specified, are usually clear from the context.

For any vector $\boldsymbol{x} \in \mathbb{F}^{np}$, we sometimes partition the vector into p blocks, each of length n. The ith block is denoted by $\boldsymbol{x}_{[i]} = \boldsymbol{x}_{S_i} \in \mathbb{F}^n$ with $S_i = \{(i-1)n+1, \dots, in\}$. More generally, $\boldsymbol{x}_{[S]}$ denotes the vector in $\mathbb{F}^{n|S|}$ formed by the blocks of lengths n indexed by the set S.

We use upper case letters such as A, B, C to represent matrices in $\mathbb{F}^{m\times n}$. The jth column of A is denoted by A_j, and the $ij$the element by A_{ij} or $A_{i,j}$. To avoid confusion with the component index, we use superscript A^α to parameterize a collection of matrices. Suppose $S \subset \{1, \dots, m\}$ and $T \subset \{1, \dots, n\}$ are index sets, then A_T denotes the submatrix of A formed by the columns of A in T, and $A_{S,T}$ the submatrix formed by the rows indexed by S and columns indexed by T. For A with np columns, $A_{[j]}$ is the submatrix of columns $\{(j-1)n+1, \dots, jn\}$; if A also has mq rows, then $A_{[i],[j]}$ is the submatrix of rows $\{(i-1)m, \dots, im\}$ and columns $\{(j-1)n+1, \dots, jn\}$. These notations can also be extended to index-set case $A_{[S]}$ and $A_{[S],[T]}$ in a natural way.

Symbols $\boldsymbol{O}$ and I_N are reserved for the zeros matrix and the identity matrix, respectively. The identity matrix usually has a subscript to indicate its dimension.

The vectorization operator $\mathrm{vec}(X) = [X_1^T\ X_2^T\ \dots\ X_p^T]^T$ stacks the columns of $X \in \mathbb{F}^{n\times p}$ into a long vector. Its inverse operator $\mathrm{mat}_{n,p}(\boldsymbol{x})$ satisfies $\mathrm{vec}(\mathrm{mat}_{n,p}(\boldsymbol{x})) = \boldsymbol{x}$.

The Kronecker product is denoted by $\otimes$. The transpose, conjugate transpose, inverse, and pseudo inverse of underlying vectors and/or matrices when appropriate are denoted respectively by ^{T}, ^{H}, $^{-1}$, and †. The trace of a square matrix A is $\mathrm{trace}(A) = \sum_i A_{ii}$, and the determinant is denoted by $\det(A)$. For two symmetric matrices A and B, $A \succeq B$ means $A - B$ is positive semidefinite and $A \succ B$ means $A - B$ is positive definite.

The ith largest singular values of a matrix A is usually denoted by $\sigma_i(A)$, and the singular value vector by $\boldsymbol{\sigma}(X) = [\sigma_1(X)\ \sigma_2(X)\ \cdots]^T$. The singular value decomposition (SVD) of an $m \times n$ real or complex matrix A is a factorization of the form

$$A = U\Sigma V^H, \tag{2.1}$$

where U is an $m \times m$ real or complex unitary matrix, Σ is an $m \times n$ rectangular diagonal matrix with the singular value vector on the diagonal, and V is an $n \times n$ real or complex unitary matrix.

A linear operator $\mathcal{A} : \mathbb{R}^{n_1 \times n_2} \mapsto \mathbb{R}^m$ can be represented by m matrices $\mathscr{A} = \{A^1, A^2, \ldots, A^m\} \subset \mathbb{R}^{n_1 \times n_2}$ as follows

$$\mathcal{A}(X) = \boldsymbol{A}\mathrm{vec}(X) \overset{\mathrm{def}}{=} \begin{bmatrix} \mathrm{vec}(A^1)^T \\ \mathrm{vec}(A^2)^T \\ \vdots \\ \mathrm{vec}(A^m)^T \end{bmatrix} \mathrm{vec}(X). \tag{2.2}$$

We will interchangeably use $\mathcal{A}$, $\mathscr{A}$ and $\boldsymbol{A}$ to represent the same linear operator.

For any linear operator $\mathcal{A} : \mathbb{R}^{n_1 \times n_2} \mapsto \mathbb{R}^m$, its adjoint operator $\mathcal{A}^* : \mathbb{R}^m \mapsto \mathbb{R}^{n_1 \times n_2}$ is defined by the following relation

$$\langle \mathcal{A}(X), \boldsymbol{z} \rangle = \langle X, \mathcal{A}^*(\boldsymbol{z}) \rangle, \quad \forall X \in \mathbb{R}^{n_1 \times n_2}, \boldsymbol{z} \in \mathbb{R}^m. \tag{2.3}$$

2.1.2 Norms

From now on, the field $\mathbb{F}$ is either $\mathbb{R}$ or $\mathbb{C}$. For any vector $\boldsymbol{x} \in \mathbb{F}^n$, the ℓ_q norms for $1 \leq q \leq \infty$ are denoted by

$$\|\boldsymbol{x}\|_q = \left(\sum_{i=1}^{n} |x_i|^q \right)^{1/q}, \ 1 \leq q < \infty \tag{2.4}$$

and

$$\|\boldsymbol{x}\|_\infty = \max_{1 \leq i \leq n} |x_i|, \ q = \infty. \tag{2.5}$$

The canonical inner product in $\mathbb{F}^n$ is defined by $\langle \boldsymbol{x}, \boldsymbol{y} \rangle = \boldsymbol{x}^H \boldsymbol{y}$. Clearly, the ℓ_2 (or Euclidean) norm is $\|\boldsymbol{x}\|_2 = \sqrt{\langle \boldsymbol{x}, \boldsymbol{x} \rangle}$.

The norm $\|\boldsymbol{x}\|_{k,1}$ is the summation of the absolute values of the k (absolutely) largest components of $\boldsymbol{x}$. In particular, the ℓ_∞ norm $\|\boldsymbol{x}\|_\infty = \|\boldsymbol{x}\|_{1,1}$ and the ℓ_1 norm $\|\boldsymbol{x}\|_1 = \|\boldsymbol{x}\|_{n,1}$. We use $\|\cdot\|_\diamond$ to denote a general vector norm.

For any $\boldsymbol{x} \in \mathbb{F}^{np}$ with p blocks, each of length n, the block-ℓ_q norms for $1 \leq q \leq \infty$ associated with this block structure are defined as:

$$\|\boldsymbol{x}\|_{\mathrm{b}q} = \left(\sum_{i=1}^{p} \|\boldsymbol{x}_{[i]}\|_2^q \right)^{1/q}, \ 1 \leq q < \infty \tag{2.6}$$

and

$$\|\boldsymbol{x}\|_{\mathrm{b}\infty} = \max_{1 \leq i \leq p} \|\boldsymbol{x}_{[i]}\|_2, \ q = \infty. \tag{2.7}$$

Obviously, the block-ℓ_2 norm is the same as the ordinary ℓ_2 norm.

The support of $\boldsymbol{x}$, $\mathrm{supp}(\boldsymbol{x})$, is the index set of the non-zero components of $\boldsymbol{x}$. The size of the support, usually denoted by the ℓ_0 "norm" $\|\boldsymbol{x}\|_0$, is the sparsity level of $\boldsymbol{x}$. The block support of $\boldsymbol{x} \in \mathbb{F}^{np}$, $\mathrm{bsupp}(\boldsymbol{x}) = \{i : \|\boldsymbol{x}_{[i]}\|_2 \neq 0\}$, is the index set of the non-zero blocks of $\boldsymbol{x}$. The size of the block support, denoted by the block-ℓ_0 "norm" $\|\boldsymbol{x}\|_{\mathrm{b}0}$, is the block-sparsity level of $\boldsymbol{x}$.

Suppose $X \in \mathbb{F}^{n_1 \times n_2}$. Define the Frobenius norm of X as $\|X\|_{\mathrm{F}} = \sqrt{\sum_{i,j} |X_{ij}|^2} = \sqrt{\sum_i \sigma_i^2(X)}$, the nuclear norm as $\|X\|_* = \sum_i \sigma_i(X)$, and the operator norm as $\|X\|_2 = \max\{\sigma_i(X)\}$, where $\sigma_i(X)$ is the ith singular value of X. The rank of X is denoted by $\mathrm{rank}(X) = |\{i : \sigma_i(X) \neq 0\}|$. The inner product of two matrices $X_1, X_2 \in \mathbb{F}^{n_1 \times n_2}$ is defined as

$$\left\langle X^1, X^2 \right\rangle = \mathrm{trace}(X^{1H} X^{2H}) = \mathrm{vec}(X^1)^H \mathrm{vec}(X^2) = \sum_{i,j} X_{ij}^1 X_{ij}^2. \tag{2.8}$$

Using the singular value vector $\boldsymbol{\sigma}(X)$, we have the following relations:

$$\begin{aligned}\|X\|_{\mathrm{F}} &= \|\boldsymbol{\sigma}(X)\|_2, \\ \|X\|_* &= \|\boldsymbol{\sigma}(X)\|_1, \\ \|X\|_2 &= \|\boldsymbol{\sigma}(X)\|_\infty, \\ \operatorname{rank}(X) &= \|\boldsymbol{\sigma}(X)\|_0. \end{aligned} \tag{2.9}$$

Note that we use $\|\cdot\|_2$ to represent both the matrix operator norm and the ℓ_2 norm of a vector. The exact meaning can always be inferred from the context.

2.1.3 Inequalities

Various inequalities will be used in the dissertation. We summarize some of the most important ones in this section. The following *Höler's inequality* is well-known:

$$\sum_{i=1}^n a_i b_i \leq \left\{\sum_{i=1}^n a_i^p\right\}^{1/p} \left\{\sum_{i=1}^n b_i^q\right\}^{1/q}, a_i \geq 0, b_i \geq 0, 1/p + 1/q = 1. \tag{2.10}$$

When $p = q = 2$, Hölder's inequality is also called *Cauchy-Schwartz inequality.*

The *Cr-inequality* that holds for $a_i \geq 0$ is also very useful:

$$\left\{\sum_{i=1}^n a_i\right\}^r \leq \begin{cases} n^{r-1}\left\{\sum_{i=1}^n a_i^r\right\}, & r \geq 1; \\ \sum_{i=1}^n a_i^r, & 0 \leq r \leq 1. \end{cases} \tag{2.11}$$

An application of the Cr-inequality to $a_i = |x_i|^p$ and $r = p/q$ gives

$$\|\boldsymbol{x}\|_p \geq \|\boldsymbol{x}\|_q, \ 0 < p \leq q \leq \infty. \tag{2.12}$$

As a consequence of the inequality (2.12) and Hölder's inequality, we obtain

$$\|\boldsymbol{x}\|_\infty \leq \|\boldsymbol{x}\|_2 \leq \|\boldsymbol{x}\|_1 \leq \sqrt{k}\|\boldsymbol{x}\|_2 \leq k\|\boldsymbol{x}\|_\infty, \tag{2.13}$$

where $k = \|\boldsymbol{x}\|_0$ is the sparsity level of $\boldsymbol{x}$, and

$$\|X\|_2 \;\leq\; \|X\|_{\mathrm{F}} \leq \|X\|_* \leq \sqrt{\mathrm{rank}(X)}\|X\|_{\mathrm{F}} \leq \mathrm{rank}(X)\|X\|_2, \tag{2.14}$$

when applied to the singular vector. The involvement of the sparsity level and the rank in the inequalities (2.13) and (2.14) makes them extremely useful in characterizing the error vectors of convex relaxation algorithms enforcing sparsity, block-sparsity, and low-rankness.

The *dual norm* $\|\cdot\|_\diamond^*$ of any given norm $\|\cdot\|_\diamond$ is defined by

$$\|\boldsymbol{x}\|_\diamond^* \;=\; \sup\{\langle \boldsymbol{x}, \boldsymbol{z}\rangle : \|\boldsymbol{z}\|_\diamond \leq 1\}. \tag{2.15}$$

Any two norms dual to each other satisfy the Cauchy-Schwartz type inequality:

$$\langle \boldsymbol{x}, \boldsymbol{z}\rangle \leq \|\boldsymbol{x}\|_\diamond \|\boldsymbol{z}\|_\diamond^*. \tag{2.16}$$

Hölder's inequality is a powerful tool to derive dual norms. We list dual norm pairs useful to this work:

$$\|\cdot\|_p \;\leftrightarrow\; \|\cdot\|_q, \tag{2.17}$$

$$\|\cdot\|_{\mathrm{b}p} \;\leftrightarrow\; \|\cdot\|_{\mathrm{b}q}, \tag{2.18}$$

$$\|\cdot\|_2 \;\leftrightarrow\; \|\cdot\|_*, \tag{2.19}$$

$$\|\cdot\|_{\mathrm{F}} \;\leftrightarrow\; \|\cdot\|_{\mathrm{F}}, \tag{2.20}$$

where $1/p + 1/q = 1$. As a consequence of the dual norm pairs, we have the following Cauchy-Schwartz inequality:

$$\begin{aligned} \left\langle X^1, X^2 \right\rangle &\leq \|X^1\|_{\mathrm{F}} \|X^2\|_{\mathrm{F}}, \\ \left\langle X^1, X^2 \right\rangle &\leq \|X^1\|_* \|X^2\|_2. \end{aligned}$$

2.2 Optimization

One of the primary contributions of this work is the design of efficient algorithms to compute performance measures with convergence guarantees. Tools from optimization and mathematical programming are thus essential. We present relevant results in this section.

2.2.1 Introduction

Mathematical optimization/programing aims at optimizing (maximizing or minimizing) an objective function over a constraint set. In its most abstract form, a *mathematical program* or an *optimization problem* is formulated as

$$\text{optimize } f(\boldsymbol{x}) \ \text{ s.t. } \ \boldsymbol{x} \in C, \tag{2.21}$$

where "s.t." is short for "subject to", $f : C \to \mathbb{R}$ is the objective function, and C is the constraint set. For a specific problem the "optimize" operation is usually replaced with more explicit min or max operations. For a minimization problem, the set of optimal solutions (optimizers, minimizers) and the optimal objective value (optimum, minimum) are usually denoted respectively by

$$\text{argmin}_{\boldsymbol{x} \in C} f(\boldsymbol{x}), \ \text{and} \ f^\star = \min_{\boldsymbol{x} \in C} f(\boldsymbol{x}). \tag{2.22}$$

If there are multiple optimal solutions, we usually use $\boldsymbol{x}^\star$ to denote any one of them and abuse notation to write $\boldsymbol{x}^\star = \text{argmin}_{\boldsymbol{x} \in C} f(\boldsymbol{x})$. The maximization case can be defined similarly.

A general *nonlinear programming* problem is given in the following more explicit form by specifying the constraint set using equality constraints and inequality constraints:

$$\min f(\boldsymbol{x}) \ \text{ s.t. } \ \boldsymbol{g}(\boldsymbol{x}) \leq 0, \ \boldsymbol{h}(\boldsymbol{x}) = 0, \ \boldsymbol{x} \in D, \tag{2.23}$$

where $D \subseteq \mathbb{R}^n$ is the domain of the nonlinear programming, $\boldsymbol{g} : \mathbb{R}^n \to \mathbb{R}^m$ and $\boldsymbol{h} : \mathbb{R}^n \to \mathbb{R}^p$ are general nonlinear functions.

Convex optimization is arguably the most important subfield of mathematical optimization, due to its wide applicability in practice and the existence of efficient algorithms with established convergence properties. In convex optimization, we assume the objective function to be minimized (resp. maximized) is convex (resp. concave), and the constraint set C is also convex. This requires f and the components of $\boldsymbol{g}$ in (2.23) are convex, and $\boldsymbol{h}$ is affine. See [43] for a gentle introduction to convex optimization.

2.2.2 Berge's Maximum Theorem

One question of interest to us is the continuity of the optimal solution and the optimal objective value with respect to parameter changes in the problem formulation. *Berge's maximum theorem* provides conditions guaranteeing such continuity:

Theorem 2.2.1. [44] *Suppose X and Θ are metric spaces, $f : X \times \Theta \to \mathbb{R}$ is jointly continuous, and the parameterized constraint $C : \Theta \twoheadrightarrow X$ is a compact-valued correspondence. Define*

$$f^\star(\boldsymbol{\theta}) = \max\{f(\boldsymbol{x}, \boldsymbol{\theta}) : \boldsymbol{x} \in C(\boldsymbol{\theta})\}, \tag{2.24}$$

$$C^\star(\boldsymbol{\theta}) = \text{argmax}\{f(\boldsymbol{x}, \boldsymbol{\theta}) : \boldsymbol{x} \in C(\boldsymbol{\theta})\}. \tag{2.25}$$

*If the correspondence C is continuous (*i.e.*, both upper and lower hemicontinous) at some $\boldsymbol{\theta} \in \Theta$, then $f^\star$ is continuous at $\boldsymbol{\theta}$ and $C^\star$ is non-empty, compact-valued, and upper hemicontinuous at $\boldsymbol{\theta}$.*

2.2.3 Lagrange Duality

For the nonlinear programming problem in (2.23) with the domain D having non-empty interior, the *Lagrangian function* $\mathcal{L} : \mathbb{R}^n \times \mathbb{R}^m_+ \times \mathbb{R}^p \to \mathbb{R}$ is defined as

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = f(\boldsymbol{x}) + \langle \boldsymbol{\lambda}, \boldsymbol{g} \rangle + \langle \boldsymbol{\nu}, \boldsymbol{h} \rangle \,. \tag{2.26}$$

The vectors $\boldsymbol{\lambda}$ and $\boldsymbol{\nu}$ are called the dual variables or *Lagrange multipliers.* The *Lagrange dual function* $\ell(\boldsymbol{\lambda}, \boldsymbol{\nu}) : \mathbb{R}^m \times \mathbb{R}^p \to \mathbb{R} \bigcup \{-\infty\}$ is defined as

$$\ell(\boldsymbol{\lambda}, \boldsymbol{\nu}) \quad = \quad \inf_{\boldsymbol{x} \in D} \mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}). \tag{2.27}$$

The dual function is always concave, even when the initial problem is not convex. And the dual problem yields lower bound on the optimal value $f^\star$ of the original problem (usually called the *primal problem*), namely, $\ell^\star \leq f^\star$. Since there exist efficient algorithms to maximize a concave function such as $\ell(\boldsymbol{\lambda}, \boldsymbol{\nu})$ if its function value, gradient, and/or Hessian are easy to compute, the dual problem provides a general framework to compute a lower bound on the optimal value of the primal problem. This procedure is usually called *Lagrange relaxation.*

If certain constraint qualification is satisfied and the original problem is convex, then we have strong duality, namely, $\ell^\star = f^\star$. When f and g_is, the components of $\boldsymbol{g}$, are convex and $\boldsymbol{h} = A\boldsymbol{x} - \boldsymbol{b}$ is affine, one such constraint qualification is the *Slater condition*: there exists an $\boldsymbol{x}$ in the relative interior of D such that

$$\boldsymbol{g}_i(\boldsymbol{x}) < 0, i = 1, \ldots, m, \; A\boldsymbol{x} = \boldsymbol{b}. \tag{2.28}$$

Note the dual norm defined in (2.15) can be viewed as the dual function of the following optimization problem:

$$\max \; 0 \;\; \text{s.t.} \;\; \boldsymbol{x} \leq 0, \boldsymbol{x} \in D = \{\boldsymbol{x} : \|\boldsymbol{x}\|_\diamond \leq 1\}, \tag{2.29}$$

since

$$\ell(\boldsymbol{\lambda}) = \sup\nolimits_{\boldsymbol{x} \in D} \mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}) = \sup\nolimits_{\boldsymbol{x} \in D} \langle \boldsymbol{\lambda}, \boldsymbol{x} \rangle = \|\boldsymbol{\lambda}\|_\diamond^*. \tag{2.30}$$

2.2.4 Karush-Kuhn-Tucker (KKT) conditions

The KKT conditions are necessary for a solution of (2.23) to be optimal. Suppose $\boldsymbol{x}^\star$ and $(\boldsymbol{\lambda}^\star, \boldsymbol{\nu}^\star)$ are primal and dual optimal points with zero duality gap. Then $\boldsymbol{x}^\star$ and $(\boldsymbol{\lambda}^\star, \boldsymbol{\nu}^\star)$ must satisfy:

- **Stationarity:**

$$\nabla f(\boldsymbol{x}^\star) + \langle \boldsymbol{\lambda}^\star, \nabla \boldsymbol{g}(\boldsymbol{x}^\star)\rangle + \langle \boldsymbol{\nu}^\star, \nabla \boldsymbol{h}(\boldsymbol{x}^\star)\rangle = 0 \tag{2.31}$$

- **Primal feasibility:**

$$\begin{aligned} \boldsymbol{g}(\boldsymbol{x}^\star) &\leq 0 \\ \boldsymbol{h}(\boldsymbol{x}^\star) &= 0 \end{aligned} \tag{2.32}$$

- **Dual feasibility:**

$$\boldsymbol{\lambda}^\star \geq 0 \tag{2.33}$$

- **Complementary slackness:**

$$\lambda_i^\star g_i(\boldsymbol{x}^\star) = 0, \; i = 1, \ldots, m. \tag{2.34}$$

If f and g_i in the primal program (2.23) are convex and $\boldsymbol{h}$ is affine, the the KKT conditions are also sufficient for $\boldsymbol{x}^\star$ and $(\boldsymbol{\lambda}^\star, \boldsymbol{\nu}^\star)$ to be primal and dual optimal.

2.2.5 Linear, Second-Order Cone, and Semidefinite Programs

Three convex optimizations that are extensively used in this disertation are linear programming, second-order cone programming, and semidefinite programming.

In a *linear program*, the objective function is linear and both the inequality and equality constraint are affine. A general linear program has the form:

$$\begin{aligned} \min \quad & \boldsymbol{c}^T \boldsymbol{x} \\ \text{s.t.} \quad & F\boldsymbol{x} \leq \boldsymbol{g} \\ & A\boldsymbol{x} = \boldsymbol{b}, \end{aligned} \tag{2.35}$$

whose dual program is

$$\begin{aligned} \max \quad & -\boldsymbol{g}^T \boldsymbol{\lambda} - \boldsymbol{b}^T \boldsymbol{\nu} \\ \text{s.t.} \quad & F^T \boldsymbol{\lambda} + A^T \boldsymbol{\nu} + \boldsymbol{c} = 0. \end{aligned} \tag{2.36}$$

In a *standard form* linear program, the only inequalities are componentwise nonnegativity constraints:

$$\begin{aligned} \min \quad & \boldsymbol{c}^T \boldsymbol{x} \\ \text{s.t.} \quad & A\boldsymbol{x} = \boldsymbol{b} \\ & \boldsymbol{x} \geq 0. \end{aligned} \tag{2.37}$$

The dual program of (2.37) is

$$\begin{aligned} \max \quad & -\boldsymbol{b}^T \boldsymbol{\nu} \\ \text{s.t.} \quad & -A^T \boldsymbol{\nu} \leq \boldsymbol{c}, \end{aligned} \tag{2.38}$$

a linear program in *inequality form.*

A *second-order cone program (SOCP)* has linear objectives and second-order cone constraints as well as linear constraints:

$$\begin{aligned} \min \quad & \boldsymbol{f}^T \boldsymbol{x} \\ \text{s.t.} \quad & \|A^i \boldsymbol{x} + \boldsymbol{b}_i\|_2 \leq \boldsymbol{c}_i^T \boldsymbol{x} + d_i, i = 1, \ldots, m \\ & F\boldsymbol{x} = \boldsymbol{g}, \end{aligned} \tag{2.39}$$

where $\boldsymbol{x} \in \mathbb{R}^n$ is the optimization variable, $A^i \in \mathbb{R}^{n_i \times n}$, and $F \in \mathbb{R}^{p \times n}$. The constraint of the form

$$\|A\boldsymbol{x} + \boldsymbol{b}\|_2 \leq \boldsymbol{c}^T \boldsymbol{x} + d \tag{2.40}$$

is called a second-order cone constraint, therefore the name SOCP. The dual of the SOCP is yet another SOCP:

$$\begin{aligned} \max \quad & -\sum_{i=1}^{m} (\boldsymbol{b}_i^T \boldsymbol{z}_i + d_i w_i) + \boldsymbol{g}^T \boldsymbol{\nu} \\ \text{s.t.} \quad & \sum_{i=1}^{m} (A^{iT} \boldsymbol{z}_i + \boldsymbol{c}_i w_i) + F^T \boldsymbol{\nu} = \boldsymbol{f} \\ & \|\boldsymbol{z}_i\|_2 \leq w_i, i = 1, \ldots, m. \end{aligned} \tag{2.41}$$

In a *semidefinite program*, besides the linear constraints and the linear objective function, the nonlinear constraints are specified by the cone of positive semidefinite matrices. More explicitly, a semidefinite program has the form

$$\begin{aligned} \min \quad & \boldsymbol{c}^T \boldsymbol{x} \\ \text{s.t.} \quad & x_1 F^1 + \cdots + x_n F^n + G \preceq 0 \\ & A\boldsymbol{x} = \boldsymbol{b}, \end{aligned} \tag{2.42}$$

where $G, F_1, \ldots, F_n$ are symmetric positive semidefinite matrices of size $k \times k$. The dual program to (2.42) is

$$\begin{aligned} \max \quad & -\boldsymbol{\nu}^T \boldsymbol{b} + \mathrm{trace}(GZ) \\ \text{s.t.} \quad & \mathrm{trace}(F^i Z) + c_i + A_i^T \boldsymbol{\nu} = 0, \ i = 1, \ldots, n \\ & Z \succeq 0. \end{aligned} \tag{2.43}$$

2.2.6 Subdifferential and Nonsmooth Optimization

Most of the low-dimensional signal recovery algorithms involve continuous but nondifferentiable functions. We gather some concepts and results in nonsmooth optimization that are useful to the analysis of these algorithms.

A vector $\boldsymbol{g}$ is a *subgradient* of a function $f : \mathbb{R}^n \to \mathbb{R}$ at $\boldsymbol{x}$ if

$$f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \boldsymbol{g}^T(\boldsymbol{y} - \boldsymbol{x}), \forall \boldsymbol{y} \in \mathbb{R}^n. \tag{2.44}$$

The set of all subgradient of f at $\boldsymbol{x}$ is called the *subdifferential* of f at $\boldsymbol{x}$, denoted by $\partial f(\boldsymbol{x})$:

$$\partial f(\boldsymbol{x}) = \{\boldsymbol{g} : f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \boldsymbol{g}^T(\boldsymbol{y} - \boldsymbol{x}), \forall \boldsymbol{y} \in \mathbb{R}^n\}. \tag{2.45}$$

The definitions of subgradient and subdifferential do not require the convexity of f. If f is convex, $\partial f(\boldsymbol{x})$ is non-empty at every point in the relative interior of the domain of f; if f is differentiable, then the subdifferential coincides with the ordinary gradient $\partial f(\boldsymbol{x}) = \{\nabla f(\boldsymbol{x})\}$.

The subdifferential of a norm $f(\boldsymbol{x}) = \|\boldsymbol{x}\|_\diamond$ on $\mathbb{R}^n$ takes the form

$$\partial\|\boldsymbol{x}\|_\diamond = \begin{cases} \{\boldsymbol{g} \in \mathbb{R}^n : \langle \boldsymbol{g}, \boldsymbol{x}\rangle = \|\boldsymbol{x}\|_\diamond, \|\boldsymbol{g}\|_\diamond^* = 1\} & \text{if } \boldsymbol{x} \neq \boldsymbol{0}; \\ \{\boldsymbol{g} \in \mathbb{R}^n : \|\boldsymbol{g}\|_\diamond^* = 1\}, & \text{if } \boldsymbol{x} = \boldsymbol{0}. \end{cases} \tag{2.46}$$

Applying this formula to $\|\cdot\|_1$, $\|\cdot\|_{\text{b1}}$, and $\|\cdot\|_*$, we have the following:

$$\begin{aligned} \partial\|\boldsymbol{x}\|_1 &= \left\{ \boldsymbol{g} \in \mathbb{R}^n : g_i = \begin{cases} 1, & \text{if } x_i > 0; \\ -1, & \text{if } x_i < 0; \\ a, |a| \leq 1 & \text{if } x_i = 0. \end{cases} \right\}, & (2.47) \\ \partial\|\boldsymbol{x}\|_{\text{b1}} &= \left\{ \boldsymbol{g} \in \mathbb{R}^{np} : \boldsymbol{g}_{[i]} = \begin{cases} \frac{\boldsymbol{x}_{[i]}}{\|\boldsymbol{x}_{[i]}\|_2}, & \text{if } \boldsymbol{x}_{[i]} \neq 0; \\ \boldsymbol{a}, \|\boldsymbol{a}\|_2 \leq 1 & \text{if } \boldsymbol{x}_{[i]} = 0. \end{cases} \right\}, & (2.48) \\ \partial\|X\|_* &= \{UV^T + W : \|W\|_2 \leq 1, U^T W = 0, WV = 0\}, & (2.49) \end{aligned}$$

where $X = U\Sigma V^T$ is the SVD of X.

The optimality conditions for both unconstrained and constrained optimizations with differentiable objective functions and constraints can be generalized to nondifferentiable cases using subdifferential. More explicitly, for unconstrained optimization with a convex objective function f, we have

$$f(\boldsymbol{x}^\star) = \min_{\boldsymbol{x}} f(\boldsymbol{x}) \Leftrightarrow \boldsymbol{0} \in \partial f(\boldsymbol{x}^\star). \tag{2.50}$$

For constrained optimization, under Slater's condition, the KKT condition is generalized to the following convex optimization

$$\min f(\boldsymbol{x}) \quad \text{s.t.} \quad \boldsymbol{g}(\boldsymbol{x}) \leq 0 \tag{2.51}$$

as: $\boldsymbol{x}^\star$ is primal optimal and $\boldsymbol{\lambda}^\star$ is dual optimal iff

$$\begin{aligned} & \boldsymbol{g}(\boldsymbol{x}^\star) \leq 0, & (2.52) \\ & \boldsymbol{\lambda}^\star \geq 0, & (2.53) \\ & 0 \in \partial f(\boldsymbol{x}^\star) + \sum_{i=1}^n \lambda_i^\star \partial g_i(\boldsymbol{x}^\star), & (2.54) \\ & \lambda_i^\star g_i(\boldsymbol{x}^\star) = 0, i = 1, \ldots, n. & (2.55) \end{aligned}$$

2.2.7 Fixed Point Iteration

A *fixed point* of a given function $\boldsymbol{f}(\boldsymbol{x}) : \mathbb{R}^n \to \mathbb{R}^n$ is a solution to $\boldsymbol{x} = \boldsymbol{f}(\boldsymbol{x})$. Given a point $\boldsymbol{x}_0$ in the domain of $\boldsymbol{f}$, the *fixed point iteration*

$$\boldsymbol{x}_{t+1} = \boldsymbol{f}(\boldsymbol{x}_t), t = 0, 1, 2, \ldots \tag{2.56}$$

which generate a sequence $\{\boldsymbol{x}_0, \boldsymbol{x}_1, \boldsymbol{x}_2, \ldots\}$. If this sequence converges to $\boldsymbol{x}^\star$ and $\boldsymbol{f}$ is continuous, then $\boldsymbol{x}^\star$ is a fixed point of $\boldsymbol{f}$, *i.e.* , $\boldsymbol{x}^\star = \boldsymbol{f}(\boldsymbol{x}^\star)$.

Many well-known algorithms can be viewed as fixed point iterations. Examples include the power method [45] and its variants to compute eigenvectors, Newton's method [46] to find roots of functions, Runge-Kutta methods [47] to approximate solutions of ordinary differential equations, some of the "successive approximation" schemes used in dynamic programming to solve Bellman's equation [48], and so on. In the power method, the function is $\boldsymbol{f}(\boldsymbol{x}) = \frac{A\boldsymbol{x}}{\|A\boldsymbol{x}\|}$. Under certain assumptions, the fixed point iteration or the power iteration

$$\boldsymbol{x}_{t+1} = \boldsymbol{f}(\boldsymbol{x}_t) = \frac{A\boldsymbol{x}_t}{\|A\boldsymbol{x}_t\|}, \ t = 0, 1, 2, \ldots \tag{2.57}$$

converges to the eigenvector corresponding to the largest eigenvalue of A. Newton's method finds the root of a scalar differentiable function $f(x)$ by using the iteration

$$x_{t+1} = x_t - \frac{f(x_t)}{f'(x_t)}, \tag{2.58}$$

which is a fixed point iteration corresponding to the function $g(x) = x - f(x)/f'(x)$. The fixed points of $g(x)$ satisfying $x = g(x)$, or equivalently, $f(x)/f'(x) = 0$ are indeed roots of $f(x)$. Newton's method can be generalized to vector functions by appropriately replacing the derivative function $f'(x)$ with the gradient matrix.

We are concerned with the existence and uniqueness of the fixed point of a scalar function defined over $\mathbb{R}_+$. One very simple sufficient condition for the existence of a fixed point over the interval $[a, b]$ for a continuous function f is $f(a) > a$ and $f(b) < b$ as ensured by the intermediate value theorem.

The following theorem establishes a similar result without the continuity condition:

Theorem 2.2.2. [49] *Suppose f is an increasing function from $\mathbb{R}_+$ to $\mathbb{R}_+$ such that $f(a) > a$ for some positive scalar a, and $f(b) < b$ for some scalr $b > a$. Then f has a positive fixed point.*

Uniqueness can usually be argued using the concavity of the function f.

2.3 Probability

One important component of this dissertation is the analysis of the proposed goodness measures' probabilistic behavior when the sensing matrices are random. In this section, we collect definitions and tools useful to the probabilistic analysis.

2.3.1 Notations

Probability and expectation operations are denoted by $\mathbb{P}$ and $\mathbb{E}$, respectively. The abbreviation *i.i.d.* represents identically and independently distributed. The Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance matrix Σ is denoted by $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$.

For a scalar random variable x, the Orlicz ψ_2 norm is defined as

$$\|x\|_{\psi_2} = \inf\left\{t > 0 : \mathbb{E}\exp\left(\frac{|x|^2}{t^2}\right) \leq 2\right\}. \tag{2.59}$$

Markov's inequality immediately gives that x with finite $\|x\|_{\psi_2}$ has a subgaussian tail

$$\mathbb{P}(|x| \geq t) \leq 2\exp(-ct^2/\|x\|_{\psi_2}). \tag{2.60}$$

The converse is also true, *i.e.*, if x has a subgaussian tail $\exp(-t^2/K^2)$, then $\|x\|_{\psi_2} \leq cK$.

A random vector $\boldsymbol{\xi} \in \mathbb{R}^n$ is called *isotropic and subgaussian* with parameter L if $\mathbb{E}|\left\langle \boldsymbol{\xi}, \boldsymbol{u} \right\rangle|^2 = \|\boldsymbol{u}\|_2^2$ and $\|\left\langle \boldsymbol{\xi}, \boldsymbol{u} \right\rangle\|_{\psi_2} \leq L\|\boldsymbol{u}\|_2$ hold for any $\boldsymbol{u} \in \mathbb{R}^n$. A random vector $\boldsymbol{\xi}$

with independent subgaussian entries $\xi_1, \ldots, \xi_n$ is a subgaussian vector because [50]

$$\begin{aligned} \|\langle \boldsymbol{\xi}, \boldsymbol{u} \rangle\|_{\psi_2} &\leq c\sqrt{\sum_{i=1}^{n} u_i^2 \|\xi_i\|_{\psi_2}^2} \\ &\leq c \max_{1\leq i\leq n} \|x_i\|_{\psi_2} \|\boldsymbol{u}\|_2. \end{aligned} \tag{2.61}$$

Clearly, if in addition $\{\xi_i\}$ are centered and has unit variance, then $\boldsymbol{\xi}$ is also isotropic. In particular, the standard Gaussian vector on $\mathbb{R}^n$ and the sign vector with *i.i.d.* 1/2 Bernoulli entries are isotropic and subgaussian. Isotropic and subgaussian random vectors also include the vectors with the normalized volume measure on various convex symmetric bodies , for example, the unit balls of ℓ_p^n for $2 \leq p \leq \infty$ [51].

2.3.2 Random Sensing Ensembles

Several types of random sensing matrices are of particular interest. We always assume the matrix A is of size $m \times n$ in this subsection. The *Gaussian ensemble* consists of matrices $A \in \mathbb{R}^{m\times n}$ whose entries follow *i.i.d.* $\mathcal{N}(0, \frac{1}{m})$. Matrices in the *Bernoulli ensemble* have *i.i.d.* entries taking values $\pm\frac{1}{\sqrt{m}}$ with probability $\pm\frac{1}{2}$. If the rows of A are normalized rows of the Fourier transform matrix randomly selected with replacement, we say A is a sample from the *Fourier ensemble.* A matrix A is called an *isotropic and subgaussian* matrix if its rows are *i.i.d.* isotropic and subgaussian random vectors. As a consequence of the discussion at the end of the previous section, the Gaussian ensemble and the Bernoulli ensemble are special cases of the isotropic and subgaussian ensemble. Random matrices from the Gaussian, Bernoulli, and Fourier ensembles are shown in Figure 2.1.

2.3.3 Gaussian Process

A Gaussian process is a stochastic process $\{\xi_t\}_{t\in T}$ indexed by the set T for which any finite linear combination of $\xi_{t_1}, \ldots, \xi_{t_n}$ follows Gaussian distributions. We also assume that ξ_t is of mean zero. The Gaussian process is very important for the probabilistic analysis not only for Gaussian random matrices, but also for general

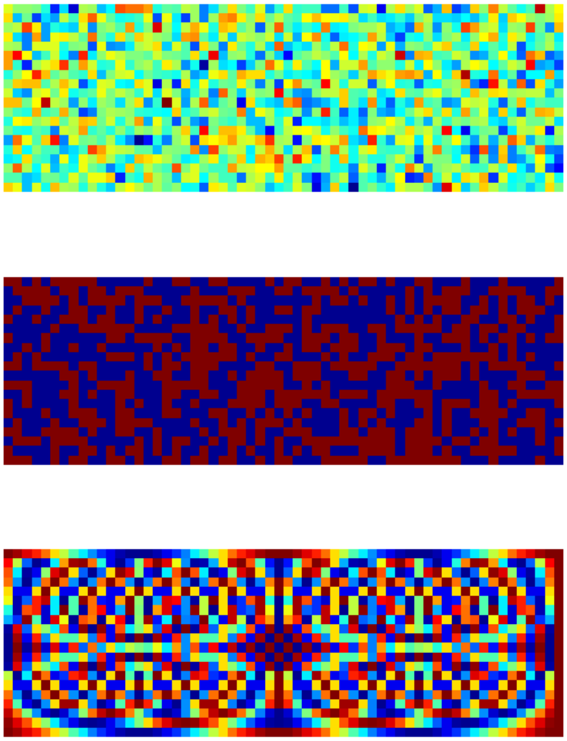


Figure 2.1: Random sensing matrices. Top to bottom: Gaussian matrix, Bernoulli matrix, real part of a Fourier matrix.

isotropic and subgaussian matrices. As we will see in Section 2.3.4, the behavior of empirical processes involving general isotropic and subgaussian random vectors can be characterized by associated Gaussian processes.

One of the most important tools for Gaussian processes is the comparison theorem, in particular the Gordon's inequality and the Slepian's inequality. Under certain conditions on the second order moments of the increments, these two inequalities compare the expected extremal values of a Gaussian process with another Gaussian process, one that is simpler for analysis purposes.

Lemma 2.3.1. [52, Chapter 3.1] *Suppose* $(\xi_{u,v})_{u\in U, v\in V}$ *and* $(\zeta_{u,v})_{u\in U, v\in V}$ *be Gaussian processes with zero mean. If for all* $u, u' \in U$ *and* $v, v' \in V$*,*

$$\mathbb{E}(\xi_{u,v} - \xi_{u',v'})^2 \leq \mathbb{E}(\zeta_{u,v} - \zeta_{u',v'})^2, \tag{2.62}$$

then

$$\textit{Slepian's inequality: } \mathbb{E}\text{sup}_{u\in U, v\in V}\xi_{u,v} \leq \mathbb{E}\text{sup}_{u\in U, v\in V}\zeta_{u,v}. \tag{2.63}$$

If

$$\begin{aligned} &\mathbb{E}(\xi_{u,v} - \xi_{u',v'})^2 \leq \mathbb{E}(\zeta_{u,v} - \zeta_{u',v'})^2, \quad \textit{if} \quad u \neq u' \\ &\mathbb{E}(\xi_{u,v} - \xi_{u,v'})^2 = \mathbb{E}(\zeta_{u,v} - \zeta_{u,v'})^2, \end{aligned} \tag{2.64}$$

then

$$\textit{Gordon's inequality: } \mathbb{E} \inf_{u\in U} \text{sup}_{v\in V} X_{u,v} \geq \mathbb{E} \inf_{u\in U} \text{sup}_{v\in V} Y_{u,v}. \tag{2.65}$$

Consider only one index set and assume the index is time. The Slepian's inequality states that, the Gaussian process with the bigger step size measured by the second order moments of the increments will have a larger maximal distance away from the origin in its life. The Gordon's inequality can be understood in a similar manner.

Suppose $G \in \mathbb{R}^{n_1 \times n_2}$ is a Gaussian matrix and $\boldsymbol{g} \sim \mathcal{N}(0, \mathrm{I}_n)$, the following estimates are useful:

$$\mathbb{E}\ \|G\|_2 \leq \sqrt{n_1} + \sqrt{n_2}, \tag{2.66}$$

$$\mathbb{E}\ \|\boldsymbol{g}\|_2 \leq \sqrt{\mathbb{E}\ \|\boldsymbol{g}\|_2^2} = \sqrt{n}, \tag{2.67}$$

$$\mathbb{E}\|\boldsymbol{g}\|_\infty = \mathbb{E} \max_i \boldsymbol{g}_i \leq \sqrt{\log n}, \tag{2.68}$$

where for (2.66) we used an upper bound for the expected largest singular value of a rectangular Gaussian matrix [53, 54], (2.67) is due to Jensen's inequality for convex functions, and (2.68) is given by [52, Equation 3.13, page 79].

A related estimate is $\mathbb{E}\|\boldsymbol{g}\|_{\mathrm{b}\infty}$ for $\boldsymbol{g} \sim \mathcal{N}(0, \mathrm{I}_{np})$, which can be bounded using the Slepian's inequality (2.63). We rearrange the vector $\boldsymbol{g} \in \mathbb{R}^{np}$ into a matrix $G \in \mathbb{R}^{n \times p}$ such that the vectorization $\mathrm{vec}(G) = \boldsymbol{g}$. Clearly, we have $\|\boldsymbol{g}\|_{\mathrm{b}\infty} = \|G\|_{1,2}$, where $\|\cdot\|_{1,2}$ denotes the matrix norm as an operator from $(\mathbb{R}^n, \|\cdot\|_{\ell_1})$ to $(\mathbb{R}^p, \|\cdot\|_{\ell_2})$. Recognizing $\|G\|_{1,2} = \max_{\boldsymbol{v} \in S^{n-1}, \boldsymbol{w} \in T^{p-1}} \langle G\boldsymbol{v}, \boldsymbol{w} \rangle$, we define the Gaussian process $X_{\boldsymbol{v},\boldsymbol{w}} = \langle G\boldsymbol{v}, \boldsymbol{w} \rangle$ indexed by $(\boldsymbol{v}, \boldsymbol{w}) \in S^{n-1} \times T^{p-1}$. Here $S^{n-1} = \{\boldsymbol{v} \in \mathbb{R}^n : \|\boldsymbol{v}\|_2 = 1\}$ and $T^{p-1} = \{\boldsymbol{w} \in \mathbb{R}^p : \|\boldsymbol{w}\|_1 = 1\}$. We compare $X_{\boldsymbol{v},\boldsymbol{w}}$ with another Gaussian process $Y_{\boldsymbol{v},\boldsymbol{w}} = \langle \boldsymbol{\xi}, \boldsymbol{v} \rangle + \langle \boldsymbol{\zeta}, \boldsymbol{w} \rangle, (\boldsymbol{v}, \boldsymbol{w}) \in S^{n-1} \times T^{p-1}$, where $\boldsymbol{\xi} \sim \mathcal{N}(0, \mathrm{I}_n)$ and $\boldsymbol{\zeta} \sim \mathcal{N}(0, \mathrm{I}_p)$. The Gaussian processes $X_{\boldsymbol{v},\boldsymbol{w}}$ and $Y_{\boldsymbol{v},\boldsymbol{w}}$ satisfy the conditions for the Slepian's inequality (See the proof of [55, Theorem 32, page 23]). Therefore, we have

$$\begin{aligned}
\mathbb{E}\|\boldsymbol{g}\|_{\mathrm{b}\infty} &= \mathbb{E} \max_{(\boldsymbol{v},\boldsymbol{w}) \in S^{n-1} \times T^{p-1}} X_{\boldsymbol{v},\boldsymbol{w}} \leq \mathbb{E} \max_{(\boldsymbol{v},\boldsymbol{w}) \in S^{n-1} \times T^{p-1}} Y_{\boldsymbol{v},\boldsymbol{w}} \\
&= \mathbb{E} \max_{\boldsymbol{v} \in S^{n-1}} \langle \boldsymbol{\xi}, \boldsymbol{v} \rangle + \mathbb{E} \max_{\boldsymbol{w} \in T^{p-1}} \langle \boldsymbol{\zeta}, \boldsymbol{w} \rangle \\
&= \mathbb{E}\|\boldsymbol{\xi}\|_2 + \mathbb{E}\|\boldsymbol{\zeta}\|_\infty \\
&\leq \sqrt{n} + \sqrt{\log p}.
\end{aligned} \tag{2.69}$$

Here we have used (2.67) and (2.68).

2.3.4 Estimates for Empirical Processes

Suppose $\{\boldsymbol{\xi}_i\}_{i=1}^{\infty}$ are *i.i.d.* random vectors. Denote P_n the *empirical measure* that puts equal mass at each of the n random observations $\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_n$, *i.e.*,

$$P_n(\cdot) = \frac{1}{n} \sum_{i=1}^{n} \delta_{\boldsymbol{\xi}_i}(\cdot) \tag{2.70}$$

with $\delta_{\boldsymbol{\xi}}(\cdot)$ the dirac measure that puts unit mass at $\boldsymbol{\xi}$. Suppose $\mathcal{F}$ is a collection of measurable functions, then the *empirical process* indexed by the function class $\mathcal{F}$ is of the form

$$\{P_n(f^2)\}_{f \in \mathcal{F}} = \left\{ \frac{1}{n} \sum_{i=1}^{n} f^2(\boldsymbol{\xi}_i) \right\}_{f \in \mathcal{F}} . \tag{2.71}$$

We have abused notation to use $P(f^2)$ to denote the integration $\int f^2 \mathrm{d}P$ with respect to a measure P.

An important result established in [51] is an estimate of the converge rate of

$$\sup_{f \in \mathcal{F}} \left| P_n(f^2) - \mathbb{E} f^2 \right| . \tag{2.72}$$

Before we turn to the general empirical process result of [51] developed by delicate use of the powerful generic chaining idea, we need some notations and definitions. A key concept in studying general Gaussian processes as well as the empirical process $\{P_m(f^2)\}_{f^2 \in \mathcal{F}}$ is the γ_p functional we are going to define. We need some setup first. For any set $\mathcal{X}$, an *admissible sequence* is a sequence of increasing partitions $\{\mathcal{Q}_k\}_{k \geq 0}$ of $\mathcal{X}$ such that $|\mathcal{Q}_0| = 1$ and $|\mathcal{Q}_k| = 2^{2^k}$ for $k \geq 1$. By a sequence of increasing partitions, we mean that every set in $\mathcal{Q}_k$ is contained in some set of $\mathcal{Q}_{k+1}$. We will use $Q_k(X)$ to denote the unique set in the partition $\mathcal{Q}_k$ that contains $X \in \mathcal{X}$. The diameter of $Q_k(X)$ is denoted by $\Delta(Q_k(X))$. Then we have the following definition for γ_p functional associated with a metric space:

Definition 2.3.1. *Suppose* $(\mathcal{X}, d)$ *is a metric space and* $p > 0$. *We define*

$$\gamma_p(\mathcal{X}, d) \quad = \quad \inf \sup_{X \in \mathcal{X}} \sum_{k \geq 0} 2^{k/p} \Delta(Q_k(X)), \tag{2.73}$$

where the infimum is taken over all admissible sequences.

The importance of the γ_p functional lies in its relationship with the behavior of a Gaussian process indexed by a metric space when the metric coincides with the one induced by the Gaussian process. More precisely, suppose $\{\xi_X\}_{X\in\mathcal{X}}$ is a Gaussian process indexed by the metric space $(\mathcal{X}, d)$ with

$$d(X, Z) = (\mathbb{E}(\xi_X - \xi_Z)^2)^{1/2}, \tag{2.74}$$

then we have

$$c\gamma_2(\mathcal{X}, d) \leq \mathbb{E}\mathrm{sup}_{X\in\mathcal{X}}\xi_X \leq C\gamma_2(\mathcal{X}, d) \tag{2.75}$$

for some numerical constants c and C. The upper bound was first established by Fernique [56] and the lower bound is obtained by Talagrand using majorizing measures [57]. The rather difficult concept of majorizing measures has been considerably simplified through the notion of "generic chaining", an idea that dates back to Kolmogorov and is greatly advanced in recently years by Talagrand [58]. The upper bound (generic chaining bound)

$$\mathbb{E}\mathrm{sup}_{X\in\mathcal{X}}\xi_X \leq C\gamma_2(\mathcal{X}, d) \tag{2.76}$$

is actually applicable as long as the increments of $\{\xi_X\}_{X\in\mathcal{X}}$ have subgaussian tails:

$$\mathbb{P}\{|\xi_X - \xi_Z| > t\} \leq c\exp\left(-\frac{t^2}{2d(X, Z)^2}\right), \quad \forall t > 0 \tag{2.77}$$

for $d(X, Z)$ defined in (2.74) and

$$\mathbb{E}\xi_X = 0, \quad \forall X \in \mathcal{X}. \tag{2.78}$$

Under the conditions (2.77) and (2.78), an immediate consequence of the generic chaining bound is the well-known *Dudley's inequality* [52, 58]

$$\mathbb{E}\mathrm{sup}_{X\in\mathcal{X}}\xi_X \leq C\sum_{k\geq 0} 2^{k/2}e_k(\mathcal{X}), \tag{2.79}$$

or equivalently in the more familiar integral form

$$\mathbb{E}\mathrm{sup}_{X\in\mathcal{X}}\xi_X \le C\int_0^\infty \sqrt{\log N(\mathcal{X}, d, \epsilon)}, \tag{2.80}$$

where $e_k(\mathcal{X})$ and $N(\mathcal{X}, d, \epsilon)$ are the entropy number and covering number [59, 60], respectively. In general the generic chaining bound (2.76) is tighter than the Dudley's entropy bounds (2.79) and (2.80).

Now we are ready to present the result on the behavior of an empirical process $\left\{\frac{1}{n}\sum_{i=1}^n f^2(\boldsymbol{\xi}_i)\right\}_{f\in\mathcal{F}}$ established in [51]:

Theorem 2.3.1. [51] *Let* $\{\boldsymbol{\xi}, \boldsymbol{\xi}_i, i = 1, \dots, m\} \subset \mathbb{R}^n$ *be* i.i.d. *random vectors which induce a measure* μ *on* $\mathbb{R}^n$*, and* $\mathcal{F}$ *be a subset of the unit sphere of* $L_2(\mathbb{R}^n, \mu)$ *with* $\mathrm{diam}(\mathcal{F}, \|\cdot\|_{\psi_2}) \overset{\text{def}}{=} \max_{f,g\in\mathcal{F}} \|f - g\|_{\psi_2} = \alpha$*. Then there exist absolute constants* c_1, c_2, c_3 *such that for any* $\epsilon > 0$ *and* $m \ge 1$ *satisfying*

$$m \ge c_1 \frac{\alpha^2\gamma_2^2(\mathcal{F}, \|\cdot\|_{\psi_2})}{\epsilon^2}, \tag{2.81}$$

with probability at least $1 - \exp(-c_2\epsilon^2 m/\alpha^4)$*,*

$$\mathrm{sup}_{f\in\mathcal{F}}\left|\frac{1}{m}\sum_{k=1}^m f^2(\boldsymbol{\xi}_k) - \mathbb{E}f^2(\boldsymbol{\xi})\right| \le \epsilon. \tag{2.82}$$

Furthermore, if $\mathcal{F}$ *is symmetric, we have*

$$\begin{aligned}&\mathbb{E}\mathrm{sup}_{f\in\mathcal{F}}\left|\frac{1}{m}\sum_{k=1}^m f^2(\boldsymbol{\xi}_k) - \mathbb{E}f^2(\boldsymbol{\xi})\right| \\ &\le c_3 \max\left\{\alpha\frac{\gamma_2(\mathcal{F}, \|\cdot\|_{\psi_2})}{\sqrt{m}}, \frac{\gamma_2^2(\mathcal{F}, \|\cdot\|_{\psi_2})}{m}\right\}.\end{aligned} \tag{2.83}$$

A case that is of particular interest to us is when $\mathcal{H}$ is a subset of the unit sphere of $\mathbb{R}^n$, $\mathcal{F} = \{\langle \boldsymbol{u}, \cdot\rangle : \boldsymbol{u} \in \mathcal{H}\}$, and $\{\boldsymbol{\xi}, \boldsymbol{\xi}_i, i = 1, \dots, m\}$ are *i.i.d.* isotropic and subgaussian random vectors with parameter L. In this case, due to (2.75), we have $\gamma_2(\mathcal{F}, \|\cdot\|_{\psi_2}) \sim \ell_*(\mathcal{H})$ with the ℓ_*-functional defined below:

Definition 2.3.2. *Let* $\mathcal{H} \subset \mathbb{R}^n$ *and* $\boldsymbol{g} \sim \mathcal{N}(0, \mathrm{I}_n)$*. Denote by* $\ell_*(\mathcal{H}) = \mathbb{E}\sup_{\boldsymbol{u}\in\mathcal{H}} \langle \boldsymbol{g}, \boldsymbol{u}\rangle$.

With these preparations, we combine [51, Theorem D] and the equivalence $\gamma_2(\mathcal{F}, \psi_2) \sim \ell_*(\mathcal{H})$ to obtain:

Theorem 2.3.2. *Let* $\{\boldsymbol{\xi}, \boldsymbol{\xi}_i, i = 1, \dots, m\} \subset \mathbb{R}^n$ *be* i.i.d. *isotropic and subgaussian random vectors,* $\mathcal{H}$ *be a subset of the unit sphere of* $\mathbb{R}^n$*, and* $\mathcal{F} = \{f_{\boldsymbol{u}}(\cdot) = \langle \boldsymbol{u}, \cdot \rangle : \boldsymbol{u} \in \mathcal{H}\}$*. Suppose* $\mathrm{diam}(\mathcal{F}, \|\cdot\|_{\psi_2}) = \alpha$*. Then there exist absolute constants* c_1, c_2, c_3 *such that for any* $\epsilon > 0$ *and* $m \geq 1$ *satisfying*

$$m \geq c_1 \frac{\alpha^2 \ell_*^2(\mathcal{H})}{\epsilon^2}, \tag{2.84}$$

with probability at least $1 - \exp(-c_2 \epsilon^2 m / \alpha^4)$*,*

$$\mathrm{sup}_{f \in \mathcal{F}} \left| \frac{1}{m} \sum_{k=1}^{m} f^2(\boldsymbol{\xi}_k) - \mathbb{E} f^2(\boldsymbol{\xi}) \right| \leq \epsilon. \tag{2.85}$$

Furthermore, if $\mathcal{F}$ *is symmetric, we have*

$$\begin{aligned} &\mathbb{E}\mathrm{sup}_{f \in \mathcal{F}} \left| \frac{1}{m} \sum_{k=1}^{m} f^2(\boldsymbol{\xi}_k) - \mathbb{E} f^2(\boldsymbol{\xi}) \right| \\ &\leq c_3 \max\left\{ \alpha \frac{\ell_*(\mathcal{H})}{\sqrt{m}}, \frac{\ell_*^2(\mathcal{H})}{m} \right\}. \end{aligned} \tag{2.86}$$

Chapter 3

Sparsity Recovery: Background

In this chapter, we present background knowledge on sparsity recovery.

3.1 Introduction to Sparsity Recovery

Sparse signal reconstruction aims at recovering a sparse signal $\boldsymbol{x} \in \mathbb{R}^n$ from observations of the following model[1]:

$$\boldsymbol{y} = A\boldsymbol{x} + \boldsymbol{w}, \tag{3.1}$$

where $A \in \mathbb{R}^{m \times n}$ is the measurement or sensing matrix, $\boldsymbol{y}$ is the measurement vector, and $\boldsymbol{w} \in \mathbb{R}^m$ is the noise vector. The sparsity level k of $\boldsymbol{x}$ is defined as the number of non-zero components of $\boldsymbol{x}$. The measurement system is underdetermined because the number of measurements m is much smaller than the signal dimension n. However, when the sparsity level k is also small, it is possible to recover $\boldsymbol{x}$ from $\boldsymbol{y}$ in a stable manner. Reconstruction of a sparse signal from linear measurements appears in many signal processing branches, such as compressive sensing [1, 11, 61], sparse linear regression [62], source localization [12, 18], sparse approximation, and signal denoising [63]. Model (3.1) is applicable to many practical areas such as DNA microarrays [64], radar imaging [65], cognitive radio [66], and sensor arrays [12, 18], to name a few.

---

[1] More often than not, the signal $\boldsymbol{x}$ is sparse when represented using a known basis or dictionary Φ, namely, $\boldsymbol{x} = \Phi \boldsymbol{s}$ where $\boldsymbol{s}$ is sparse. In this case, the matrix Φ can be absorbed into the sensing matrix A. Therefore, without loss of generality, we assume $\boldsymbol{x}$ is sparse.

Research interest in sparsity recovery has been renewed in the past decade due to the introduction of a new signal acquisition scheme: Compressive Sensing. The traditional Sampling-and-Compression paradigm for data acquisition samples a signal of interest, *e.g.*, an image, and then uses compression techniques such as wavelet transform to reduce the size of the sampled signal for storage or transmission purposes. Since the majority of the sampled signal will end up being thrown away, why do we take them in the first place? Compressive Sensing combines the process of signal sampling and compression by replacing signal samples with more general linear measurements and exploiting the sparsity property of most natural signals [1, 11, 61]. Most natural signals are compressible under some basis and are well approximated by their $k-$sparse representations [67]. Therefore, this scheme, if properly justified, will reduce the necessary sampling rate beyond the limit set by Nyquist and Shannon [5, 10]. Surprisingly, for exact $k-$sparse signals, if $m = O(k \log(n/k)) \ll n$ and the measurement matrix is generated randomly from, for example, a Gaussian distribution, we can recover a $k-$sparse signal exactly in the noise-free setting by solving a linear programming task. Besides, various methods have been designed for the noisy case [68–72]. Along with these algorithms, rigorous theoretic analysis is provided to guarantee their effectiveness in terms of, for example, various l_p-norms of the recovery error [68–72].

One important technique to apply sparsity recovery algorithms in practical problems is to create an artificial sparse signal by discretization. In [12], the authors transform the process of source localization using sensory arrays into the task of estimating the spectrum of a sparse signal by discretizing the parameter manifold. For simplicity, suppose we have an array with m sensors on the plane, then at a specific time, the output vector $\boldsymbol{y} \in \mathbb{C}^m$ of the sensor array is a linear combination of array responses for k sources distinguished by k direction-of-arrivals (DOAs) $\{\theta_i\}_{i=1}^k \subset (0, \pi)$:

$$\boldsymbol{y} = \sum_{i=1}^{k} \boldsymbol{a}(\theta_i) x_i + \boldsymbol{w} = A(\boldsymbol{\theta})\boldsymbol{x} + \boldsymbol{w}. \tag{3.2}$$

Here $\boldsymbol{x} \in \mathbb{C}^k$ is vector of the unknown source signals, $\boldsymbol{w} \in \mathbb{C}^m$ is an additive noise, and the transfer matrix $A(\boldsymbol{\theta}) \in \mathbb{C}^{m \times k}$ and the vector $\boldsymbol{\theta} \in \mathbb{R}^k$ are given by

$$A(\boldsymbol{\theta}) = [\boldsymbol{a}(\theta_1) \cdots \boldsymbol{a}(\theta_k)] \text{ and } \boldsymbol{\theta} = [\theta_1 \cdots \theta_k]^T. \tag{3.3}$$

The array response vector $\boldsymbol{a}(\theta)$ depends on the array configuration. The authors of [12] discretize the parameter range $(0,\pi)$ into n grid points $\{\hat{\theta}_j\}_{j=1}^n$, and assume the true DOAs $\{\theta_i\}_{i=1}^k$ are among these grid points, *i.e.* , $\{\theta_i\}_{i=1}^k \subset \{\hat{\theta}_j\}_{j=1}^n$. Then the model (3.4) is rewritten into

$$\boldsymbol{y} = \sum_{j=1}^{n} \boldsymbol{a}(\hat{\theta}_j)\hat{x}_j + \boldsymbol{w} = A(\hat{\boldsymbol{\theta}})\hat{\boldsymbol{x}} + \boldsymbol{w}, \tag{3.4}$$

where

$$A(\hat{\boldsymbol{\theta}}) = [\boldsymbol{a}(\hat{\theta}_1)\cdots\boldsymbol{a}(\hat{\theta}_n)] \in \mathbb{C}^{m\times n},\ \hat{\boldsymbol{\theta}} = [\hat{\theta}_1\cdots\hat{\theta}_n]^T \in \mathbb{R}^n, \tag{3.5}$$

and the jth component of the $k-$sparse vector $\hat{\boldsymbol{x}}$ is

$$\hat{x}_j = \begin{cases} x_i, & \text{if } \theta_i = \hat{\theta}_j; \\ 0, & \text{otherwise.} \end{cases} \tag{3.6}$$

Therefore, the problem of estimating the DOAs $\{\theta_i\}_{i=1}^k$ for k sources is equivalent with estimating the sparse signal $\hat{\boldsymbol{x}}$. This method exhibits super-resolution in the estimation of DOA compared with traditional techniques such as beamforming [13], Capon [14], and MUSIC [15,16]. Since the basic model (3.4) applies to several other important problems in signal processing (see [17] and references therein), the principle is readily applicable to those cases. This idea is later generalized and extended to other source localization settings in [18–20] and radar problems in [21,22,73].

3.2 Recovery Algorithms

The power of sparsity recovery techniques partially comes from the fact that there exist efficient algorithms with theoretical convergence guarantees to reconstruct the sparse signal $\boldsymbol{x}$ from the measurement vector $\boldsymbol{y}$. This fact is by no means trivial. Consider the noise-free case in which we need to recover a k-sparse signal $\boldsymbol{x} \in \mathbb{R}^n$ from the vector

$$\boldsymbol{y} = A\boldsymbol{x} \in \mathbb{R}^m \tag{3.7}$$

with m much smaller than n. Without the sparsity constraint, there will be infinitely many solutions. With the k-sparsity constraint, if the matrix A is such that any $2k$ columns are linearly independent (which requires $m \geq 2k$), then the solution is unique. A plausible heuristic to obtain the k-sparse signal $\boldsymbol{x}$ is to solve the following optimization problem which minimizes the number of non-zero elements subject to the observation constraint:

$$\min_{\boldsymbol{z} \in \mathbb{R}^n} \|\boldsymbol{z}\|_0 \quad \text{s.t.} \quad A\boldsymbol{z} = A\boldsymbol{x}. \tag{3.8}$$

Apparently, $\boldsymbol{x}$ is the unique solution to (3.8) for any k-sparse vector $\boldsymbol{x}$ if and only if A is such that any of its $2k$ columns are linearly independent. Unfortunately, the combinatorial optimization problem (3.8) is NP hard. A major breakthrough in sparsity recovery is to establish that under certain mild conditions on A, one could replace the ℓ_0 "norm" with the ℓ_1 norm in (3.8) and obtain the sparse signal $\boldsymbol{x}$ by solving [1, 11, 61]:

$$\min_{\boldsymbol{z} \in \mathbb{R}^n} \|\boldsymbol{z}\|_1 \quad \text{s.t.} \quad A\boldsymbol{z} = A\boldsymbol{x}, \tag{3.9}$$

which is a convex relaxation of (3.8). The optimization problem (3.9) can be rewritten as a linear program:

$$\min_{\boldsymbol{z}, \boldsymbol{u}} \mathbf{1}_n^T \boldsymbol{u} \quad \text{s.t.} \quad \boldsymbol{z} - \boldsymbol{u} \leq 0, \; -\boldsymbol{z} - \boldsymbol{u} \leq 0, \; A\boldsymbol{z} = A\boldsymbol{x}. \tag{3.10}$$

Here $\mathbf{1}_n \in \mathbb{R}^n$ is the column vector of all ones.

In the noisy setting, many algorithms have also been proposed to recover $\boldsymbol{x}$ from $\boldsymbol{y}$ in a stable manner. We focus on three algorithms based on ℓ_1 minimization: the Basis Pursuit (BP) [74], the Dantzig selector (DS) [69], and the LASSO estimator (LASSO) [75].

$$\text{BP:} \min_{\boldsymbol{z} \in \mathbb{R}^n} \|\boldsymbol{z}\|_1 \quad \text{s.t.} \; \|\boldsymbol{y} - A\boldsymbol{z}\|_\diamond \leq \epsilon \tag{3.11}$$

$$\text{DS:} \min_{\boldsymbol{z} \in \mathbb{R}^n} \|\boldsymbol{z}\|_1 \quad \text{s.t.} \quad \|A^T(\boldsymbol{y} - A\boldsymbol{z})\|_\infty \leq \mu \tag{3.12}$$

$$\text{LASSO:} \min_{\boldsymbol{z} \in \mathbb{R}^n} \frac{1}{2}\|\boldsymbol{y} - A\boldsymbol{z}\|_2^2 + \mu\|\boldsymbol{z}\|_1. \tag{3.13}$$

Here μ is a tuning parameter, and ϵ is a measure of the noise level. In the noise-free case where $\boldsymbol{w} = 0$, roughly speaking all the three algorithms reduce to (3.9).

The BP algorithm [74] tries to minimize the ℓ_1 norm of solutions subject to the measurement constraint. It is applicable to both noiseless settings and bounded noise settings with a known noise bound ϵ. The BP was originally developed for the noise-free case, *i.e.* , $\epsilon = 0$ in (3.11). In this dissertation, we refer to both cases as the BP. In the context of sparse approximation, the BP avoids overfitting by selecting a parsimonious representation within a specified approximation error limit. For $\epsilon > 0$, (3.11) can be reformulated as a linear program or second-order cone program depending on $\diamond = 1, 2$, or ∞. For example, when $\diamond = 2$, the BP can be recast as the second-order cone program:

$$\min_{\boldsymbol{z},\boldsymbol{u}} \mathbf{1}_n^T \boldsymbol{u} \ \text{ s.t. } \ \boldsymbol{z} - \boldsymbol{u} \le 0, \ -\boldsymbol{z} - \boldsymbol{u} \le 0, \ \|A\boldsymbol{z} - \boldsymbol{y}\|_2 \le \epsilon. \tag{3.14}$$

The DS [69] aims to reconstruct a reasonable signal in most cases when the measurement is contaminated by Gaussian noise. The constraint $\|A^T(\boldsymbol{y} - A\boldsymbol{z})\|_\infty \le \mu$ requires that all feasible solutions must have uniformly bounded correlations between the induced residual vector $\boldsymbol{y} - A\boldsymbol{x}$ and the columns of the sensing matrix A. One motivation behind this constraint is to include in the solution variables that are highly correlated with the observation $\boldsymbol{y}$. Refer to [69] for discussions on reasons of controlling the size of the correlated residual vector rather than the size of the residual itself as in Basis Pursuit. We emphasize that the optimization problem (DS) is convex and can be cast as a linear program [69]:

$$\min_{\boldsymbol{u},\ \boldsymbol{z}\in\mathbb{R}^n} \mathbf{1}_n^T \boldsymbol{u} \ \text{ s.t. } \ \boldsymbol{z} - \boldsymbol{u} \le 0, \ -\boldsymbol{z} - \boldsymbol{u} \le 0, \ \ -\mu\mathbf{1}_n \le A^T(\boldsymbol{y} - A\boldsymbol{z}) \le \mu\mathbf{1}_n. \tag{3.15}$$

The LASSO estimator, as originally introduced, solved the optimization problem [75]:

$$\min_{\boldsymbol{z}\in\mathbb{R}^n} \frac{1}{2}\|\boldsymbol{y} - A\boldsymbol{z}\|_2^2 \ \text{ s.t. } \ \|\boldsymbol{z}\|_1 \le t, \tag{3.16}$$

for some $t > 0$. Following the convention in [76], in this dissertaion we refer to the solution to the closely related optimization problem (3.13) as the LASSO estimator. The optimization problems (3.16) and (3.13) are equivalent in the sense that given

$t \geq 0$, there exists a $\mu \geq 0$ such that the two problems have the same solution, and vice versa [77]. Problem (3.16) is usually referred to as constrained regression, while problem (3.13) is ℓ_1-penalized regression. Both programs can be explicitly rewritten as standard second-order cone programs and solved using a primal-dual log-barrier algorithm [78, 79].

3.3 Null Space Property and Restricted Isometry Property

In the noise-free case, a minimal requirement on the convex relaxation algorithm (3.9) is the *uniqueness and exactness* of the solution $\hat{\boldsymbol{x}} \overset{\text{def}}{=} \text{argmin}_{\boldsymbol{z}:A\boldsymbol{z}=A\boldsymbol{x}} \|\boldsymbol{x}\|_1$, *i.e.*, $\hat{\boldsymbol{x}} = \boldsymbol{x}$ for any k-sparse signal $\boldsymbol{x}$. The sufficient and necessary condition for unique and exact ℓ_1 recovery is given by the Null Space Property (NSP) [34–36]:

$$\|\boldsymbol{z}_S\|_1 < \|\boldsymbol{z}_{S^c}\|_1, \forall \boldsymbol{z} \in \text{null}(A),\ |S| \leq k, \tag{3.17}$$

or equivalently

$$\|\boldsymbol{z}\|_{k,1} < \frac{1}{2}\|\boldsymbol{z}\|_1, \forall \boldsymbol{z} \in \text{null}(A). \tag{3.18}$$

Note that the sufficient and necessary condition for unique and exact ℓ_0 recovery according to (3.8), any $2k$ columns of A are linearly independent, can be equivalently expressed as

$$k < \frac{1}{2}\|\boldsymbol{z}\|_0, \forall \boldsymbol{z} \in \text{null}(A), \tag{3.19}$$

or

$$\|\boldsymbol{z}_S\|_0 < \|\boldsymbol{z}_{S^c}\|_0, \forall \boldsymbol{z} \in \text{null}(A),\ |S| \leq k. \tag{3.20}$$

Therefore, the NSP can be viewed as an ℓ_1 relaxation of the sufficient and necessary condition for unique ℓ_0 recovery.

In the noisy case, many quantities have been proposed to study the recovery errors of the BP, the DS and the LASSO, for example, the Restricted Isometry Constant (RIC) [39, 61], the Restricted Eigenvalue assumption [80], and the Restricted Correlation assumption [81], among others. The most popular quantity is the RIC, which we follow [39, 61] to define as follows:

Definition 3.3.1. *For each integer $k \in \{1, \ldots, n\}$, the restricted isometry constant (RIC) δ_k of a matrix $A \in \mathbb{R}^{m\times n}$ is defined as the smallest $\delta > 0$ such that*

$$1 - \delta \leq \frac{\|A\boldsymbol{x}\|_2^2}{\|\boldsymbol{x}\|_2^2} \leq 1 + \delta \tag{3.21}$$

holds for arbitrary non-zero $k-$sparse signal $\boldsymbol{x}$.

The RIC has a very clear geometrical meaning. Roughly speaking, a matrix A with a small δ_k is nearly an isometry between Euclidean spaces when restricted onto all $k-$sparse vectors. Apparently, $\delta_{2k}(A) < 1$ if and only if any $2k$ columns of the matrix A are linearly independent. Therefore, $\delta_{2k}(A) < 1$ is the necessary and sufficient condition for exact ℓ_0 recovery. As a consequence of the error bound (6.142), which will be presented in the following, $\delta_{2k}(A) < \sqrt{2} - 1$ is a sufficient condition for the unique and exact recovery of the ℓ_1 minimization algorithm (3.9), suggesting $\delta_{2k}(A) < \sqrt{2} - 1$ implies the NSP. The converse is not true.

Now we cite some of the most renowned performance results on the BP, the DS, and the LASSO, which are expressed in terms of the RIC. Assume $\boldsymbol{x}$ is a $k-$sparse signal and $\hat{\boldsymbol{x}}$ is its estimate given by any of the three algorithms; then we have the following:

1. BP [39]: *Suppose that $\delta_{2k} < \sqrt{2} - 1$ and $\|\boldsymbol{w}\|_2 \leq \epsilon$. The solution to the BP* (3.11) *satisfies*

$$\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2 \leq \frac{4\sqrt{1+\delta_{2k}}}{1 - (1+\sqrt{2})\delta_{2k}} \cdot \epsilon. \tag{3.22}$$

2. DS [69]: *If the noise $\boldsymbol{w}$ satisfies $\|A^T\boldsymbol{w}\|_\infty < \mu$, and $\delta_{2k} + \delta_{3k} < 1$, then, the error signal obeys*

$$\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2 \leq \frac{4\sqrt{k}}{1 - \delta_{2k} - \delta_{3k}}\mu. \tag{3.23}$$

3. LASSO [82]: *If the noise* $\boldsymbol{w}$ *satisfies* $\|A^T\boldsymbol{w}\|_\infty < \mu$, *and* $\delta_{2k} < 1/(3\sqrt{2}+1)$, *then, the error signal of* (3.13) *satisfies*

$$\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2 \leq \frac{16\sqrt{k}}{(1-\delta_{2k})\left(1 - \frac{3\sqrt{2}\delta_{2k}}{1-\delta_{2k}}\right)^2}\mu. \tag{3.24}$$

We note that in these error bounds, the terms involving the RIC on the right hand sides are quite complicated expressions.

3.4 Probabilistic Analysis

Although the RIC provides a measure quantifying the goodness of a sensing matrix, its computation poses great challenge. The computational difficulty is compensated by the nice properties of the RIC for a large class of random sensing matrices. We cite one general result below [83]:

Let $A \in \mathbb{R}^{m\times n}$ *be a random matrix whose entries are* i.i.d. *samples from any distribution that satisfies the concentration inequality for any* $\boldsymbol{x} \in \mathbb{R}^n$ *and* $0 < \varepsilon < 1$*:*

$$\mathbb{P}\left(\left|\|A\boldsymbol{x}\|_2^2 - \|\boldsymbol{x}\|_2^2\right| \geq \varepsilon\|\boldsymbol{x}\|_2^2\right) \leq 2e^{-mc_0(\varepsilon)}, \tag{3.25}$$

where $c_0(\varepsilon)$ *is a constant depending only on* ε *and such that for all* $\varepsilon \in (0,1)$, $c_0(\varepsilon) > 0$. *Then, for any given* $\delta \in (0,1)$, *there exist constants* $c_1, c_2 > 0$ *depending only on* δ *such that* $\delta_k \leq \delta$, *with probability not less than* $1 - 2e^{-c_2 m}$, *as long as*

$$m \geq c_1 k \log\frac{n}{k}. \tag{3.26}$$

We remark that distributions satisfying the concentration inequality (3.25) include the Gaussian distribution and the Bernoulli distribution. To see the implication of (3.26), we suppose $n = 1,000,000$, the approximate size of a 1024×1024 image, and $k = 10\%n$, then we need roughly $m = 2.3kc_1$ Gaussian measurements to recover the original signal.

Chapter 4

Computable Performance Analysis for Sparsity Recovery

4.1 Goodness Measures and Error Bounds

In this section, we derive performance bounds on the ℓ_∞ norms of the error vectors. We first establish a theorem characterizing the error vectors for the ℓ_1 recovery algorithms BP (3.11), DS (3.12), and LASSO (3.13).

Proposition 4.1.1. *Suppose* $\boldsymbol{x}$ *in* (3.1) *is* $k-$*sparse and the noise* $\boldsymbol{w}$ *satisfies* $\|\boldsymbol{w}\|_\diamond \leq \epsilon$, $\|A^T\boldsymbol{w}\|_\infty \leq \mu$, *and* $\|A^T\boldsymbol{w}\|_\infty \leq \kappa\mu,\ \kappa \in (0,1)$, *for the BP, the DS, and the LASSO, respectively. Define* $\boldsymbol{h} = \hat{\boldsymbol{x}} - \boldsymbol{x}$ *as the error vector for any of the three* ℓ_1 *recovery algorithms* (3.11), (3.12), *and* (3.13). *Then we have*

$$c\|\boldsymbol{h}\|_{k,1} \geq \|\boldsymbol{h}\|_1, \tag{4.1}$$

where $c = 2$ *for the BP and the DS, and* $c = 2/(1-\kappa)$ *for the LASSO.*

Proof of Proposition 4.1.1. Suppose $S = \mathrm{supp}(\boldsymbol{x})$ and $|S| = \|\boldsymbol{x}\|_0 = k$. Define the error vector $\boldsymbol{h} = \hat{\boldsymbol{x}} - \boldsymbol{x}$. For any vector $\boldsymbol{z} \in \mathbb{R}^n$ and any index set $S \subseteq \{1, \ldots, n\}$, we use $\boldsymbol{z}_S \in \mathbb{R}^{|S|}$ to represent the vector whose elements are those of $\boldsymbol{z}$ indicated by S.

We first deal with the BP and the DS. As observed by Candés in [39], the fact that $\|\hat{\boldsymbol{x}}\|_1 = \|\boldsymbol{x} + \boldsymbol{h}\|_1$ is the minimum among all $\boldsymbol{z}$s satisfying the constraints in (3.11) and (3.12), together with the fact that the true signal $\boldsymbol{x}$ satisfies the constraints as required by the conditions imposed on the noise in Proposition 4.1.1, imply that

$\|\boldsymbol{h}_{S^c}\|_1$ cannot be very large. To see this, note that

$$\begin{aligned}\|\boldsymbol{x}\|_1 &\geq \|\boldsymbol{x}+\boldsymbol{h}\|_1\\ &= \sum_{i\in S}|x_i+h_i| + \sum_{i\in S^c}|x_i+h_i|\\ &\geq \|\boldsymbol{x}_S\|_1 - \|\boldsymbol{h}_S\|_1 + \|\boldsymbol{h}_{S^c}\|_1\\ &= \|\boldsymbol{x}\|_1 - \|\boldsymbol{h}_S\|_1 + \|\boldsymbol{h}_{S^c}\|_1. \qquad (4.2)\end{aligned}$$

Therefore, we obtain $\|\boldsymbol{h}_S\|_1 \geq \|\boldsymbol{h}_{S^c}\|_1$, which leads to

$$2\|\boldsymbol{h}_S\|_1 \geq \|\boldsymbol{h}_S\|_1 + \|\boldsymbol{h}_{S^c}\|_1 = \|\boldsymbol{h}\|_1. \qquad (4.3)$$

We now turn to the LASSO (3.13). We use the proof technique in [7] (see also [80]). Since the noise $\boldsymbol{w}$ satisfies $\|A^T\boldsymbol{w}\|_\infty \leq \kappa\mu$ for some small $\kappa > 0$, and $\hat{\boldsymbol{x}}$ is a solution to (3.13), we have

$$\frac{1}{2}\|A\hat{\boldsymbol{x}}-\boldsymbol{y}\|_2^2 + \mu\|\hat{\boldsymbol{x}}\|_1 \leq \frac{1}{2}\|A\boldsymbol{x}-\boldsymbol{y}\|_2^2 + \mu\|\boldsymbol{x}\|_1.$$

Consequently, substituting $\boldsymbol{y} = A\boldsymbol{x}+\boldsymbol{w}$ yields

$$\begin{aligned}\mu\|\hat{\boldsymbol{x}}\|_1 &\leq \frac{1}{2}\|A\boldsymbol{x}-\boldsymbol{y}\|_2^2 - \frac{1}{2}\|A\hat{\boldsymbol{x}}-\boldsymbol{y}\|_2^2 + \mu\|\boldsymbol{x}\|_1\\ &= \frac{1}{2}\|\boldsymbol{w}\|_2^2 - \frac{1}{2}\|A(\hat{\boldsymbol{x}}-\boldsymbol{x})-\boldsymbol{w}\|_2^2 + \mu\|\boldsymbol{x}\|_1\\ &= \frac{1}{2}\|\boldsymbol{w}\|_2^2 - \frac{1}{2}\|A(\hat{\boldsymbol{x}}-\boldsymbol{x})\|_2^2\\ &\qquad + \langle A(\hat{\boldsymbol{x}}-\boldsymbol{x}), \boldsymbol{w}\rangle - \frac{1}{2}\|\boldsymbol{w}\|_2^2 + \mu\|\boldsymbol{x}\|_1\\ &\leq \langle A(\hat{\boldsymbol{x}}-\boldsymbol{x}), \boldsymbol{w}\rangle + \mu\|\boldsymbol{x}\|_1\\ &= \left\langle \hat{\boldsymbol{x}}-\boldsymbol{x}, A^T\boldsymbol{w}\right\rangle + \mu\|\boldsymbol{x}\|_1.\end{aligned}$$

Using the Cauchy-Swcharz type inequality, we get

$$\begin{aligned}\mu\|\hat{\boldsymbol{x}}\|_1 &\leq \|\hat{\boldsymbol{x}}-\boldsymbol{x}\|_1\|A^T\boldsymbol{w}\|_\infty + \mu\|\boldsymbol{x}\|_1\\ &= \kappa\mu\|\boldsymbol{h}\|_1 + \mu\|\boldsymbol{x}\|_1,\end{aligned}$$

which leads to

$$\|\hat{\boldsymbol{x}}\|_1 \;\; \leq \;\; \kappa\|\boldsymbol{h}\|_1 + \|\boldsymbol{x}\|_1.$$

Therefore, similar to the argument in (4.2), we have

$$\begin{aligned}
& \|\boldsymbol{x}\|_1 \\
\geq \;& \|\hat{\boldsymbol{x}}\|_1 - \kappa\|\boldsymbol{h}\|_1 \\
= \;& \|\boldsymbol{x} + \boldsymbol{h}_{S^c} + \boldsymbol{h}_S\|_1 - \kappa\left(\|\boldsymbol{h}_{S^c} + \boldsymbol{h}_S\|_1\right) \\
\geq \;& \|\boldsymbol{x} + \boldsymbol{h}_{S^c}\|_1 - \|\boldsymbol{h}_S\|_1 - \kappa\left(\|\boldsymbol{h}_{S^c}\|_1 + \|\boldsymbol{h}_S\|_1\right) \\
= \;& \|\boldsymbol{x}\|_1 + (1-\kappa)\|\boldsymbol{h}_{S^c}\|_1 - (1+\kappa)\|\boldsymbol{h}_S\|_1,
\end{aligned}$$

where $S = \mathrm{supp}(\boldsymbol{x})$. Consequently, we have

$$\|\boldsymbol{h}_S\|_1 \;\; \geq \;\; \frac{1-\kappa}{1+\kappa}\|\boldsymbol{h}_{S^c}\|_1.$$

Therefore, similar to (4.3), we obtain

$$\begin{aligned}
\frac{2}{1-\kappa}\|\boldsymbol{h}_S\|_1 \;\; &\geq \;\; \frac{1+\kappa}{1-\kappa}\|\boldsymbol{h}_S\|_1 + \frac{1-\kappa}{1-\kappa}\|\boldsymbol{h}_S\|_1 \\
&\geq \;\; \frac{1+\kappa}{1-\kappa}\frac{1-\kappa}{1+\kappa}\|\boldsymbol{h}_{S^c}\|_1 + \frac{1-\kappa}{1-\kappa}\|\boldsymbol{h}_S\|_1 \\
&= \;\; \|\boldsymbol{h}\|_1. \qquad (4.4)
\end{aligned}$$

□

When the noise $\boldsymbol{w} \sim \mathcal{N}(0, \sigma^2 \mathrm{I}_m)$, as shown by Candés and Tao in [69], with high probability, $\boldsymbol{w}$ satisfies the orthogonality condition

$$|\boldsymbol{w}^T A_j| \;\; \leq \;\; \lambda_n \sigma \quad \text{ for all } \; 1 \leq j \leq n, \qquad (4.5)$$

for $\lambda_n = \sqrt{2\log n}$. More specifically, defining the event

$$E \overset{\text{def}}{=} \{\|A^T \boldsymbol{w}\|_\infty \leq \lambda_n \sigma\}, \qquad (4.6)$$

we have

$$\mathbb{P}(E^c) \leq \frac{2n \cdot (2\pi)^{-1/2} e^{-\lambda_n^2/2}}{\lambda_n}. \tag{4.7}$$

Therefore, with $\lambda_n = \sqrt{2(1+t)\log n}$, we obtain

$$\mathbb{P}(E) \geq 1 - \left(\sqrt{\pi(1+t)\log n} \cdot n^t\right)^{-1}. \tag{4.8}$$

As a consequence, the conditions on noise in Proposition (4.1.1) holds with high probability for $\mu = \lambda_n \sigma = \sqrt{2(1+t)\log n}\sigma$.

An immediate corollary of Proposition 4.1.1 is to bound the ℓ_1 and ℓ_2 norms of the error vector using the ℓ_∞ norm:

Corollary 4.1.1. *Under the assumptions of Proposition 4.1.1, we have*

$$\begin{aligned} \|\boldsymbol{h}\|_1 &\leq ck\|\boldsymbol{h}\|_\infty, & (4.9)\\ \|\boldsymbol{h}\|_2 &\leq \sqrt{ck}\|\boldsymbol{h}\|_\infty. & (4.10) \end{aligned}$$

Furthermore, if $S = \operatorname{supp}(\boldsymbol{x})$ *and* $\beta = \min_{i \in S} |\boldsymbol{x}_i|$, *then* $\|\boldsymbol{h}\|_\infty < \beta/2$ *implies*

$$\operatorname{supp}(\max(|\hat{\boldsymbol{x}}| - \beta/2, 0) = \operatorname{supp}(\boldsymbol{x}), \tag{4.11}$$

i.e.*, a thresholding operator recovers the signal support.*

For ease of presentation, we define the following goodness measures:

Definition 4.1.1. *For any* $s \in [1, n]$ *and matrix* $A \in \mathbb{R}^{m \times n}$, *define*

$$\omega_\diamond(Q, s) = \min_{\boldsymbol{z}: \|\boldsymbol{z}\|_1/\|\boldsymbol{z}\|_\infty \leq s} \frac{\|Q\boldsymbol{z}\|_\diamond}{\|\boldsymbol{z}\|_\infty}, \tag{4.12}$$

where Q *is either* A *or* $A^T A$.

Now we present the error bounds on the ℓ_∞ norm of the error vectors for the BP, the DS, and the LASSO .

Theorem 4.1.1. *Under the assumption of Proposition 4.1.1, we have*

$$\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_\infty \leq \frac{2\epsilon}{\omega_\diamond(A, 2k)} \tag{4.13}$$

for the BP,

$$\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_\infty \leq \frac{2\mu}{\omega_\infty(A^T A, 2k)} \tag{4.14}$$

for the DS, and

$$\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_\infty \leq \frac{(1+\kappa)\mu}{\omega_\infty(A^T A, 2k/(1-\kappa))} \tag{4.15}$$

for the LASSO .

Proof. Observe that for the BP

$$\begin{aligned} \|A(\hat{\boldsymbol{x}} - \boldsymbol{x})\|_2 &\leq \|\boldsymbol{y} - A\hat{\boldsymbol{x}}\|_2 + \|\boldsymbol{y} - A\boldsymbol{x}\|_2 \\ &\leq \epsilon + \|A\boldsymbol{w}\|_2 \\ &\leq 2\epsilon, \end{aligned} \tag{4.16}$$

and similarly,

$$\|A^T A(\hat{\boldsymbol{x}} - \boldsymbol{x})\|_\infty \leq 2\mu \tag{4.17}$$

for the DS, and

$$\|A^T A(\hat{\boldsymbol{x}} - \boldsymbol{x})\|_\infty \ \leq \ (1+\kappa)\mu \tag{4.18}$$

for the LASSO . The conclusions of Theorem 4.1.1 follow from equation (4.9) and Definition 4.1.1. □

One of the primary contributions of this work is the design of algorithms that compute $\omega_\diamond(A, s)$ and $\omega_\infty(A^T A, s)$ efficiently. The algorithms provide a way to numerically assess the performance of the BP, the DS, and the LASSO according to the bounds given in Theorem 4.1.1. According to Corollary 4.1.1, the correct recovery of signal support is also guaranteed by reducing the ℓ_∞ norm to some threshold. In Section 4.4, we also demonstrate that the bounds in Theorem 4.1.1 are non-trivial for a large class of random sensing matrices, as long as m is relatively large. Numerical simulations in Section 4.5 show that in many cases the error bounds on the ℓ_2 norms based on Corollary 4.1.1 and Theorem 4.1.1 are tighter than the RIC based bounds. We expect the bounds on the ℓ_∞ norms in Theorem 4.1.1 are even tighter, as we do not need the relaxation in Corollary 4.1.1.

We note that a prerequisite for these bounds to be valid is the positiveness of the involved $\omega_\diamond(\cdot)$. We call the validation of $\omega_\diamond(\cdot) > 0$ the verification problem. Note that from Theorem 4.1.1, $\omega_\diamond(\cdot) > 0$ implies the exact recovery of the true signal $\boldsymbol{x}$ in the noise-free case. Therefore, verifying $\omega_\diamond(\cdot) > 0$ is equivalent to verifying a sufficient condition for exact ℓ_1 recovery.

4.2 Verification Algorithm

In this section and the next section, we present algorithms for verification and computation of $\omega_\diamond(\cdot)$. We will present a very general algorithm and make it specific only when necessary. For this purpose, we use Q to denote either A or $A^T A$, and use $\|\cdot\|_\diamond$ to denote a general norm.

Verifying $\omega_\diamond(Q, s) > 0$ amounts to making sure $\|\boldsymbol{z}\|_1/\|\boldsymbol{z}\|_\infty \leq s$ for all $\boldsymbol{z}$ such that $Q\boldsymbol{z} = 0$. Equivalently, we can compute

$$s_* \;=\; \min_{\boldsymbol{z}} \frac{\|\boldsymbol{z}\|_1}{\|\boldsymbol{z}\|_\infty} \text{ s.t. } \; Q\boldsymbol{z} = 0. \tag{4.19}$$

Then, when $s < s_*$, we have $\omega_\diamond(Q, s) > 0$. We rewrite the optimization (4.19) as

$$\frac{1}{s_*} = \max_{\boldsymbol{z}} \|\boldsymbol{z}\|_\infty \text{ s.t. } \; Q\boldsymbol{z} = 0, \|\boldsymbol{z}\|_1 \leq 1, \tag{4.20}$$

which is solved using the following n linear programs:

$$\max_{\boldsymbol{z}} \boldsymbol{z}_i \text{ s.t. } \; Q\boldsymbol{z} = 0, \|\boldsymbol{z}\|_1 \leq 1. \tag{4.21}$$

The dual problem for (4.21) is

$$\min_{\boldsymbol{\lambda}} \|\mathbf{e}_i - Q^T \boldsymbol{\lambda}\|_\infty, \tag{4.22}$$

where $\mathbf{e}_i$ is the ith canonical basis vector.

We solve (4.21) using the primal-dual algorithm expounded in Chapter 11 of [43], which gives an implementation much more efficient than the one for solving its dual (4.22) in [40]. This method is also used to implement the ℓ_1 MAGIC for sparse signal recovery [79]. Due to the equivalence of $A^T A\boldsymbol{z} = 0$ and $A\boldsymbol{z} = 0$, we always solve (4.20) for $Q = A$ and avoid $Q = A^T A$. The former apparently involves solving linear programs of smaller size. In practice, we usually replace A with the matrix with orthogonal rows obtained from the economy-size QR decomposition of A^T.

As a dual of (4.22), (4.21) (and hence (4.20) and (4.19)) shares the same limitation as (4.22), namely, it verifies $\omega_\diamond > 0$ only for s up to $2\sqrt{2m}$. We now reformulate Proposition 4 of [40] in our framework:

Proposition 4.2.1. [40, Proposition 4] *For any $m \times n$ matrix A with $n \geq 32m$, one has*

$$s_* \;=\; \min\left\{\frac{\|\boldsymbol{z}\|_1}{\|\boldsymbol{z}\|_\infty} : Q\boldsymbol{z} = 0\right\} < 2\sqrt{2m}. \tag{4.23}$$

4.3 Computation Algorithm

Now we turn to the computation of $\omega_\diamond$. The optimization problem is as follows:

$$\omega_\diamond(Q, s) = \min_{\boldsymbol{z}} \frac{\|Q\boldsymbol{z}\|_\diamond}{\|\boldsymbol{z}\|_\infty} \text{ s.t. } \frac{\|\boldsymbol{z}\|_1}{\|\boldsymbol{z}\|_\infty} \leq s, \tag{4.24}$$

or equivalently,

$$\frac{1}{\omega_\diamond(Q, s)} = \max_{\boldsymbol{z}} \|\boldsymbol{z}\|_\infty \text{ s.t. } \|Q\boldsymbol{z}\|_\diamond \leq 1, \frac{\|\boldsymbol{z}\|_1}{\|\boldsymbol{z}\|_\infty} \leq s. \tag{4.25}$$

We will show that $1/\omega_\diamond(Q,s)$ is the unique fixed point of certain scalar function. To this end, we define functions $f_{s,i}(\eta), i=1,\ldots,n$ and $f_s(\eta)$ over $[0,\infty)$ parameterized by $s \in (1, s_*)$ as:

$$\begin{aligned} f_{s,i}(\eta) &\stackrel{\text{def}}{=} \max_{\boldsymbol{z}} \{\boldsymbol{z}_i : \|Q\boldsymbol{z}\|_\diamond \leq 1, \|\boldsymbol{z}\|_1 \leq s\eta\} \\ &= \max_{\boldsymbol{z}} \{|\boldsymbol{z}_i| : \|Q\boldsymbol{z}\|_\diamond \leq 1, \|\boldsymbol{z}\|_1 \leq s\eta\}, \end{aligned} \tag{4.26}$$

where the second equality is due to the symmetry of the domain for the maximization to the origin, and

$$\begin{aligned} f_s(\eta) &\stackrel{\text{def}}{=} \max_{\boldsymbol{z}} \{\|\boldsymbol{z}\|_\infty : \|Q\boldsymbol{z}\|_\diamond \leq 1, \|\boldsymbol{z}\|_1 \leq s\eta\} \\ &= \max_{\substack{\boldsymbol{z}:\|Q\boldsymbol{z}\|_\diamond \leq 1 \\ \|\boldsymbol{z}\|_1 \leq s\eta}} \max_i |\boldsymbol{z}_i| \\ &= \max_i \max_{\substack{\boldsymbol{z}:\|Q\boldsymbol{z}\|_\diamond \leq 1 \\ \|\boldsymbol{z}\|_1 \leq s\eta}} |\boldsymbol{z}_i| \\ &= \max_i f_{s,i}(\eta), \end{aligned} \tag{4.27}$$

where for the last but one equality we have exchanged the two maximizations. For $\eta > 0$, it is easy to show that strong duality holds for the optimization problem defining $f_{s,i}(\eta)$. As a consequence, we have the dual form of $f_{s,i}(\eta)$:

$$f_{s,i}(\eta) = \min_{\boldsymbol{\lambda}} s\eta \|\mathbf{e}_i - Q^T\boldsymbol{\lambda}\|_\infty + \|\boldsymbol{\lambda}\|_\diamond^*, \tag{4.28}$$

where $\|\cdot\|_\diamond^*$ is the dual norm of $\|\cdot\|_*$.

In the definition of $f_s(\eta)$, we basically replaced the $\|\boldsymbol{z}\|_\infty$ in the denominator of the fractional constraint in (4.25) with η. The following theorem states that the unique positive fixed point of $f_s(\eta)$ is exactly $1/\omega_\diamond(Q,s)$.

Theorem 4.3.1. *The functions $f_{s,i}(\eta)$ and $f_s(\eta)$ have the following properties:*

1. *$f_{s,i}(\eta)$ and $f_s(\eta)$ are continuous in η;*
2. *$f_{s,i}(\eta)$ and $f_s(\eta)$ are strictly increasing in η;*
3. *$f_{s,i}(\eta)$ is concave for every i;*

4. *$f_s(0) = 0$, $f_s(\eta) \geq s\eta > \eta$ for sufficiently small $\eta > 0$, and there exists $\rho < 1$ such that $f_s(\eta) < \rho\eta$ for sufficiently large η; the same holds for $f_{s,i}(\eta)$;*

5. *$f_{s,i}$ and $f_s(\eta)$ have unique positive fixed points $\eta_i^* = f_{s,i}(\eta_i^*)$ and $\eta^* = f_s(\eta^*)$, respectively; and $\eta^* = \max_i \eta_i^*$;*

6. *The unique fixed point of $f_s(\eta)$, η^*, is equal to $1/\omega_\diamond(Q, s)$;*

7. *For $\eta \in (0, \eta^*)$, we have $f_s(\eta) > \eta$; and for $\eta \in (\eta^*, \infty)$, we have $f_s(\eta) < \eta$; the same statement holds also for $f_{s,i}(\eta)$.*

8. *For any $\epsilon > 0$, there exists $\rho_1(\epsilon) > 1$ such that $f_s(\eta) > \rho_1(\epsilon)\eta$ as long as $0 < \eta \leq (1 - \epsilon)\eta^*$; and there exists $\rho_2(\epsilon) < 1$ such that $f_s(\eta) < \rho_2(\epsilon)\eta$ as long as $\eta > (1 + \epsilon)\eta^*$.*

Typical plots of the functions $f_s(\eta)$ and $f_{s,i}(\eta)$ are shown in Figure 4.1. All the properties except 6) are illustrated in the figure. Note that although $f_{s,i}(\eta)$s are concave, $f_s(\eta)$ is not concave as clearly indicated in the plot.

Proof. 1. Since in the optimization problem defining $f_{s,i}(\eta)$, the objective function $\boldsymbol{z}_i$ is continuous, and the constraint correspondence

$$\begin{aligned} C(\eta): \quad & [0, \infty) \twoheadrightarrow \mathbb{R}^n \\ & \eta \mapsto \{\boldsymbol{z} : \|Q\boldsymbol{z}\|_\diamond \leq 1, \|\boldsymbol{z}\|_1 \leq s\eta\} \end{aligned} \tag{4.29}$$

is compact-valued and continuous (both upper and lower hemicontinuous), according to Berge's Maximum Theorem 2.2.1, the optimal value function $f_{s,i}(\eta)$ is continuous. The continuity of $f_s(\eta)$ follows from that finite maximization preserves the continuity.

2. To show the strict increasing property, suppose $0 < \eta_1 < \eta_2$ and the dual variable $\boldsymbol{\lambda}_2^*$ achieves $f_{s,i}(\eta_2)$ in (4.28). Then we have

$$\begin{aligned} f_{s,i}(\eta_1) &\leq s\eta_1\|\mathbf{e}_i - Q^T\boldsymbol{\lambda}_2^*\|_\infty + \|\boldsymbol{\lambda}_2\|_\diamond^* \\ &< s\eta_2\|\mathbf{e}_i - Q^T\boldsymbol{\lambda}_2^*\|_\infty + \|\boldsymbol{\lambda}_2\|_\diamond^* \\ &= f_{s,i}(\eta_2). \end{aligned} \tag{4.30}$$

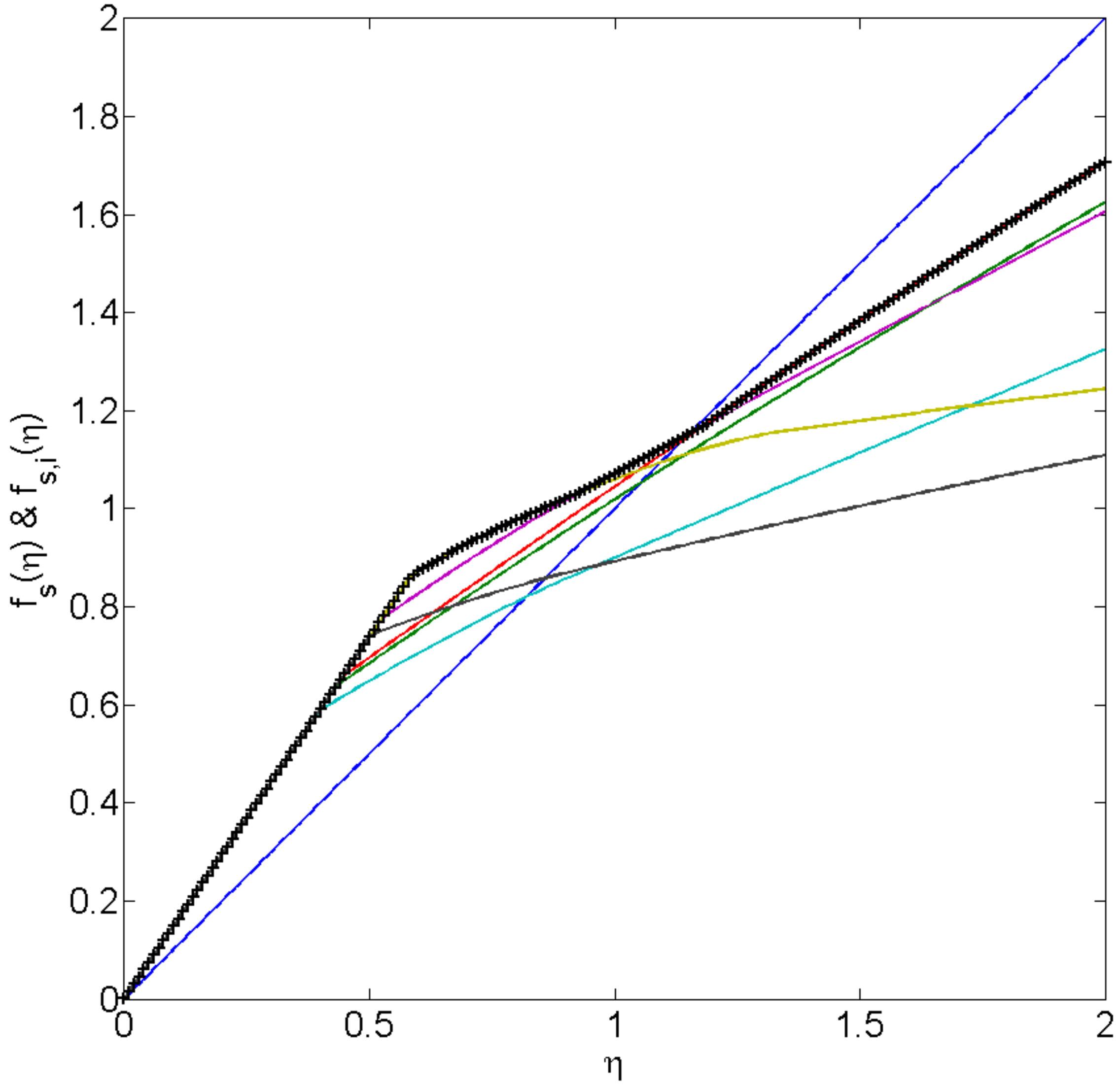


Figure 4.1: The functions $f_s(\eta)$ and $f_{s,i}(\eta)$. The blue line is for the diagonal $f = \eta$. The thick black plot is for $f_s(\eta)$, and other colored thin plots are for $f_{s,i}(\eta)$s.

The case for $\eta_1 = 0$ is proved by continuity, and the strict increasing of $f_s(\eta)$ also follows immediately.

3. The concavity of $f_{s,i}(\eta)$ follows from the dual representation (4.28) and the fact that $f_{s,i}(\eta)$ is the minimization of a function of variables η and $\boldsymbol{\lambda}$, and when $\boldsymbol{\lambda}$, the variable to be minimized, is fixed, the function is linear in η.

4. Next we show that when $\eta > 0$ is sufficiently small $f_s(\eta) \geq s\eta$. Take $\boldsymbol{z} = s\eta\mathbf{e}_i$. We have $\|\boldsymbol{z}\|_1 = s\eta$ and $\boldsymbol{z}_i = s\eta > \eta$ (recall $s \in (1, \infty)$). In addition, when $0 < \eta \leq 1/(s\|Q_i\|_\diamond)$, we also have $\|Q\boldsymbol{z}\|_\diamond \leq 1$. Therefore, for sufficiently small η, we have $f_{s,i}(\eta) \geq s\eta > \eta$. Clearly, $f_s(\eta) = \max_i f_{s,i}(\eta) \geq s\eta > \eta$ for such η.

   Recall that

$$\frac{1}{s_*} = \max_i \min_{\boldsymbol{\lambda}_i} \|\mathbf{e}_i - Q^T\boldsymbol{\lambda}_i\|_\infty. \tag{4.31}$$

   Suppose $\boldsymbol{\lambda}_i^*$ is the optimal solution for each $\min_{\boldsymbol{\lambda}_i} \|\mathbf{e}_i - Q^T\boldsymbol{\lambda}_i\|_\infty$. For each i, we then have

$$\frac{1}{s_*} \geq \|\mathbf{e}_i - Q^T\boldsymbol{\lambda}_i^*\|_\infty, \tag{4.32}$$

   which implies

$$\begin{aligned} f_{s,i}(\eta) &= \min_{\boldsymbol{\lambda}_i} s\eta\|\mathbf{e}_i - Q^T\boldsymbol{\lambda}_i\|_\infty + \|\boldsymbol{\lambda}_i\|_\diamond^* \\ &\leq s\eta\|\mathbf{e}_i - Q^T\boldsymbol{\lambda}_i^*\|_\infty + \|\boldsymbol{\lambda}_i^*\|_\diamond^* \\ &\leq \frac{s}{s_*}\eta + \|\boldsymbol{\lambda}_i^*\|_2. \end{aligned} \tag{4.33}$$

   As a consequence, we obtain

$$f_s(\eta) = \max_i f_{s,i}(\eta) \leq \frac{s}{s_*}\eta + \max_i \|\boldsymbol{\lambda}_i^*\|_2. \tag{4.34}$$

   Pick $\rho \in (s/s_*, 1)$. Then, we have the following when $\eta > \max_i \|\boldsymbol{\lambda}_i^*\|_2/(\rho - s/s_*)$:

$$\begin{aligned} f_{s,i}(\eta) &\leq \rho\eta, i = 1, \ldots, n, \text{ and} \\ f_s(\eta) &\leq \rho\eta. \end{aligned} \tag{4.35}$$

5. We first show the existence and uniqueness of the positive fixed points for $f_{s,i}(\eta)$. The properties 1) and 4) imply that $f_{s,i}(\eta)$ has at least one positive fixed point. (Interestingly, 2) and 4) also imply the existence of a positive fixed point according to Theorem 2.2.2.) To prove uniqueness, suppose there are two fixed points $0 < \eta_1^* < \eta_2^*$. Pick η_0 small enough such that $f_{s,i}(\eta_0) > \eta_0 > 0$ and $\eta_0 < \eta_1^*$. Then $\eta_1^* = \lambda\eta_0 + (1-\lambda)\eta_2^*$ for some $\lambda \in (0,1)$, which implies that $f_{s,i}(\eta_1^*) \geq \lambda f_{s,i}(\eta_0) + (1-\lambda) f_{s,i}(\eta_2^*) > \lambda\eta_0 + (1-\lambda)\eta_2^* = \eta_1^*$ due to the concavity, contradicting with $\eta_1^* = f_{s,i}(\eta_1^*)$.

   The set of positive fixed point for $f_s(\eta)$, $\{\eta \in (0,\infty) : \eta = f_s(\eta) = \max_i f_{s,i}(\eta)\}$, is a subset of $\bigcup_{i=1}^p \{\eta \in (0,\infty) : \eta = f_{s,i}(\eta)\} = \{\eta_i^*\}_{i=1}^n$. We argue that

   $$\eta^* = \max_i \eta_i^* \tag{4.36}$$

   is the unique positive fixed point for $f_s(\eta)$.

   We proceed to show that η^* is a fixed point of $f_s(\eta)$. Suppose η^* is a fixed point of $f_{s,i_0}(\eta)$, then it suffices to show that $f_s(\eta^*) = \max_i f_{s,i}(\eta^*) = f_{s,i_0}(\eta^*)$. If this is not the case, there exists $i_1 \neq i_0$ such that $f_{s,i_1}(\eta^*) > f_{s,i_0}(\eta^*) = \eta^*$. The continuity of $f_{s,i_1}(\eta)$ and the property 4) imply that there exists $\eta > \eta^*$ with $f_{s,i_1}(\eta) = \eta$, contradicting with the definition of η^*.

   To show the uniqueness, suppose η_1^* is fixed point of $f_{s,i_1}(\eta)$ satisfying $\eta_1^* < \eta^*$. Then, we must have $f_{s,i_0}(\eta_1^*) > f_{s,i_1}(\eta_1^*)$ because otherwise the continuity implies the existence of another fixed point of $f_{s,i_0}(\eta)$. As a consequence, $f_s(\eta_1^*) > f_{s,i_1}(\eta_1^*) = \eta_1^*$ and η_1^* is not a fixed point of $f_s(\eta)$.

6. Next we show $\eta^* = \gamma^* \stackrel{\text{def}}{=} 1/\omega_\diamond(Q,s)$. We first prove $\gamma^* \geq \eta^*$ for the fixed point $\eta^* = f_s(\eta^*)$. Suppose $\boldsymbol{z}^*$ achieves the optimization problem defining $f_s(\eta^*)$, *i.e.*,

   $$\eta^* = f_s(\eta^*) = \|\boldsymbol{z}^*\|_\infty, \|Q\boldsymbol{z}^*\|_\diamond \leq 1, \|\boldsymbol{z}^*\|_1 \leq s\eta^*. \tag{4.37}$$

   Since $\|\boldsymbol{z}^*\|_1/\|\boldsymbol{z}^*\|_\infty \leq s\eta^*/\eta^* \leq s$, we have

   $$\gamma^* \;\; \geq \;\; \frac{\|\boldsymbol{z}^*\|_\infty}{\|Q\boldsymbol{z}^*\|_\diamond} \geq \eta^*. \tag{4.38}$$

If $\eta^* < \gamma^*$, we define $\eta_0 = (\eta^* + \gamma^*)/2$ and

$$\begin{aligned} \boldsymbol{z}^{\mathrm{c}} &= \operatorname{argmax}_{\boldsymbol{z}} \frac{s\|\boldsymbol{z}\|_\infty}{\|\boldsymbol{z}\|_1} \text{ s.t. } \|Q\boldsymbol{z}\|_\diamond \leq 1, \|\boldsymbol{z}\|_\infty \geq \eta_0, \\ \rho &= \frac{s\|\boldsymbol{z}^{\mathrm{c}}\|_\infty}{\|\boldsymbol{z}^{\mathrm{c}}\|_1}. \end{aligned} \tag{4.39}$$

Suppose $\boldsymbol{z}^{**}$ with $\|Q\boldsymbol{z}^{**}\|_\diamond = 1$ achieves the optimum of the optimization defining $\gamma^* = 1/\omega_\diamond(Q, s)$. Clearly, $\|\boldsymbol{z}^{**}\|_\infty = \gamma^* > \eta_0$, which implies $\boldsymbol{z}^{**}$ is a feasible point of the optimization problem defining $\boldsymbol{z}^{\mathrm{c}}$ and ρ. As a consequence, we have

$$\rho \geq \frac{s\|\boldsymbol{z}^{**}\|_\infty}{\|\boldsymbol{z}^{**}\|_1} \geq 1. \tag{4.40}$$

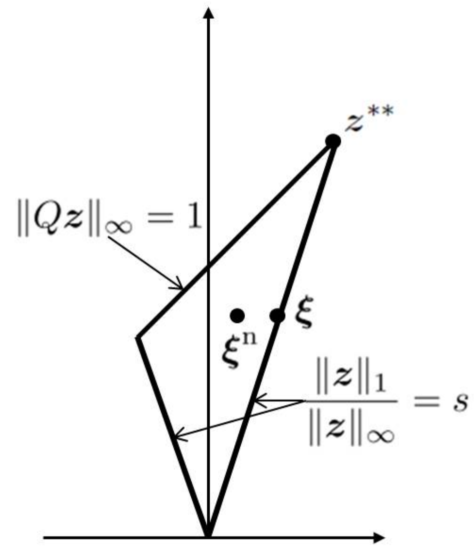

z^{**}
$\|Q\boldsymbol{z}\|_\infty = 1$
$\boldsymbol{\xi}$
$\boldsymbol{\xi}^{\mathrm{n}}$
$\frac{\|\boldsymbol{z}\|_1}{\|\boldsymbol{z}\|_\infty} = s$

Figure 4.2: Illustration of the proof for $\rho > 1$.

Actually we will show that $\rho > 1$. If $\|\boldsymbol{z}^{**}\|_1 < s\|\boldsymbol{z}^{**}\|_\infty$, we are done. If not (*i.e.*, $\|\boldsymbol{z}^{**}\|_1 = s\|\boldsymbol{z}^{**}\|_\infty$), as illustrated in Figure 4.2, we consider $\boldsymbol{\xi} = \frac{\eta_0}{\gamma^*}\boldsymbol{z}^{**}$, which satisfies

$$\|Q\boldsymbol{\xi}\|_\diamond \leq \frac{\eta_0}{\gamma^*} < 1, \tag{4.41}$$

$$\|\boldsymbol{\xi}\|_\infty = \eta_0, \text{ and} \tag{4.42}$$

$$\|\boldsymbol{\xi}\|_1 = s\eta_0. \tag{4.43}$$

To get $\boldsymbol{\xi}^{\mathrm{n}}$ as shown in Figure 4.2, pick the component of $\boldsymbol{\xi}$ with the smallest non-zero absolute value, and scale that component by a small positive constant less than 1. Because $s > 1$, $\boldsymbol{\xi}$ has more than one non-zero blocks, implying $\|\boldsymbol{\xi}^{\mathrm{n}}\|_{\mathrm{b}\infty}$ will remain the same. If the scaling constant is close enough to 1, $\|Q\boldsymbol{\xi}^{\mathrm{n}}\|_{\diamond}$ will remain less than 1. But the good news is that $\|\boldsymbol{\xi}^{\mathrm{n}}\|_1$ decreases, and hence $\rho \geq \frac{s\|\boldsymbol{\xi}^{\mathrm{n}}\|_\infty}{\|\boldsymbol{\xi}^{\mathrm{n}}\|_1}$ becomes greater than 1.

Now we proceed to obtain a contradiction that $f_s(\eta^*) > \eta^*$. If $\|\boldsymbol{z}^{\mathrm{c}}\|_1 \leq s \cdot \eta^*$, then it is a feasible point of

$$\max_{\boldsymbol{z}} \|\boldsymbol{z}\|_\infty \text{ s.t. } \|Q\boldsymbol{z}\|_\diamond \leq 1, \|\boldsymbol{z}\|_1 \leq s \cdot \eta^*. \tag{4.44}$$

As a consequence, $f_s(\eta^*) \geq \|\boldsymbol{z}^{\mathrm{c}}\|_\infty \geq \eta_0 > \eta^*$, contradicting with η^* is a fixed point and we are done. If this is not the case, *i.e.*, $\|\boldsymbol{z}^{\mathrm{c}}\|_1 > s \cdot \eta^*$, we define a new point

$$\boldsymbol{z}^{\mathrm{n}} = \tau \boldsymbol{z}^{\mathrm{c}} \tag{4.45}$$

with

$$\tau = \frac{s \cdot \eta^*}{\|\boldsymbol{z}^{\mathrm{c}}\|_1} < 1. \tag{4.46}$$

Note that $\boldsymbol{z}^{\mathrm{n}}$ is a feasible point of the optimization problem defining $f_s(\eta^*)$ since

$$\|Q\boldsymbol{z}^{\mathrm{n}}\|_\diamond = \tau\|Q\boldsymbol{z}^{\mathrm{c}}\|_\diamond < 1, \text{ and} \tag{4.47}$$

$$\|\boldsymbol{z}^{\mathrm{n}}\|_{\mathrm{b}1} = \tau\|\boldsymbol{z}^{\mathrm{c}}\|_1 = s \cdot \eta^*. \tag{4.48}$$

Furthermore, we have

$$\|\boldsymbol{z}^{\mathrm{n}}\|_\infty = \tau\|\boldsymbol{z}^{\mathrm{c}}\|_\infty = \rho\eta^*. \tag{4.49}$$

As a consequence, we obtain

$$f_s(\eta^*) \;\; \geq \;\; \rho\eta^* > \eta^*. \tag{4.50}$$

Therefore, for the fixed point η^*, we have $\eta^* = \gamma^* = 1/\omega_\diamond(Q, s)$.

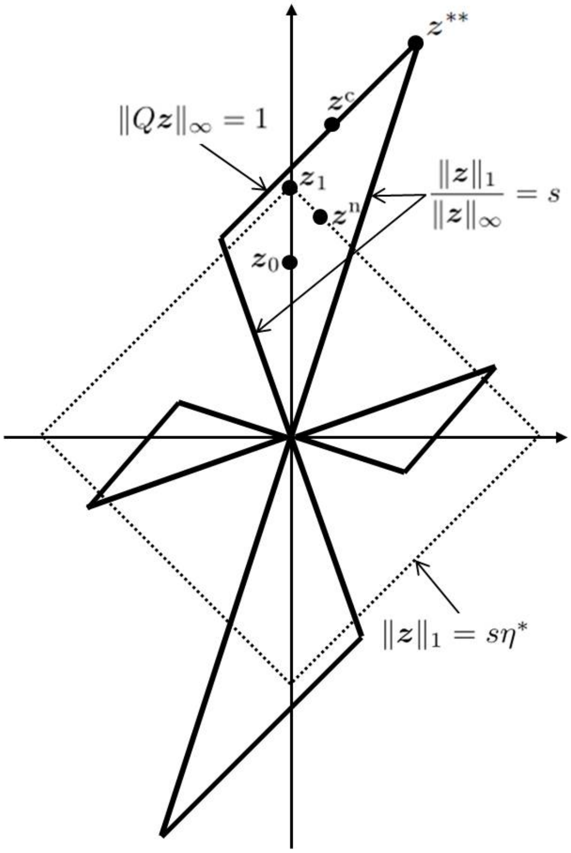


Figure 4.3: Illustration of the proof for $f_s(\eta^*) \geq \rho\eta^*$.

7. This property simply follows from the continuity, the uniqueness, and property 4).

8. We use contradiction to show the existence of $\rho_1(\epsilon)$ in 8). In view of 4), we need only to show the existence of such a $\rho_1(\epsilon)$ that works for $\eta_L \leq \eta \leq (1-\epsilon)\eta^*$ where $\eta_L = \sup\{\eta : f_s(\xi) > s\xi, \forall 0 < \xi \leq \eta\}$. Suppose otherwise, we then construct sequences $\{\eta^{(k)}\}_{k=1}^\infty \subset [\eta_L, (1-\epsilon)\eta^*]$ and $\{\rho_1^{(k)}\}_{k=1}^\infty \subset (1, \infty)$ with

$$\begin{aligned} &\lim_{k\to\infty} \rho_1^{(k)} = 1, \\ &f_s(\eta^{(k)}) \leq \rho^{(k)}\eta^{(k)}. \end{aligned} \tag{4.51}$$

Due to the compactness of $[\eta_L, (1-\epsilon)\eta^*]$, there must exist a subsequence $\{\eta^{(k_l)}\}_{l=1}^\infty$ of $\{\eta^{(k)}\}$ such that $\lim_{l\to\infty} \eta^{(k_l)} = \eta_{\lim}$ for some $\eta_{\lim} \in [\eta_L, (1-\epsilon)\eta^*]$. As a consequence of the continuity of $f_s(\eta)$, we have

$$f_s(\eta_{\lim}) = \lim_{l\to\infty} f_s(\eta^{(k_l)}) \leq \lim_{l\to\infty} \rho_1^{(k_l)}\eta^{(k_l)} = \eta_{\lim}. \tag{4.52}$$

Again due to the continuity of $f_s(\eta)$ and the fact that $f_s(\eta) < \eta$ for $\eta < \eta_L$, there exists $\eta_c \in [\eta_L, \eta_{\lim}]$ such that

$$f_s(\eta_c) \quad = \quad \eta_c, \tag{4.53}$$

contradicting with the uniqueness of the fixed point for $f_s(\eta)$. The existence of $\rho_2(\epsilon)$ can be proved in a similar manner.

□

Theorem 4.3.1 implies three ways to compute the fixed point of η^* for $f_s(\eta)$, $1/\omega_\diamond(Q,s)$.

1. **Naive Fixed-Point Iteration:** Property 8) of Theorem 4.3.1 suggests that the fixed point iteration

   $$\eta_{t+1} \quad = \quad f_s(\eta_t), t = 0, 1, \ldots \tag{4.54}$$

   starting from any initial point η_0 converges to η^*, no matter $\eta_0 < \eta^*$ or $\eta_0 > \eta^*$. The algorithm can be made more efficient in the case $\eta_0 < \eta^*$. More specifically, since $f_s(\eta) = \max_i f_{s,i}(\eta)$, at each fixed-point iteration, we set η_{t+1} to be the first $f_{s,i}(\eta_t)$ that is greater than $\eta_t + \epsilon$ with ϵ some tolerance parameter. If for all i, $f_{s,i}(\eta_t) < \eta_t + \epsilon$, then $f_s(\eta_t) = \max_i f_{s,i}(\eta_t) < \eta_t + \epsilon$, which indicates the optimal function value can not be improved greatly and the algorithm should terminate. In most cases, to get η_{t+1}, we need to solve only one optimization problem $\min_{\boldsymbol{\lambda}} s\eta\|\mathbf{e}_i - Q^T\boldsymbol{\lambda}\|_\infty + \|\boldsymbol{\lambda}\|_\diamond^*$ instead of n. This is in contrast to the case where $\eta_0 > \eta^*$, because in the later case we must compute all $f_{s,i}(\eta_t)$ to update $\eta_{t+1} = \max_i f_{s,i}(\eta_t)$. An update based on a single $f_{s,i}(\eta_t)$ might generate a value smaller than η^*.

   In Figure 4.4, we illustrate the behavior of the naive fixed-point iteration algorithm (4.54). These figures are generated by Matlab for a two dimensional problem. We index the sub-figures from left to right and from top to bottom. The first (upper left) sub-figure shows the star-shaped region $\mathcal{S} = \{\boldsymbol{z} :$

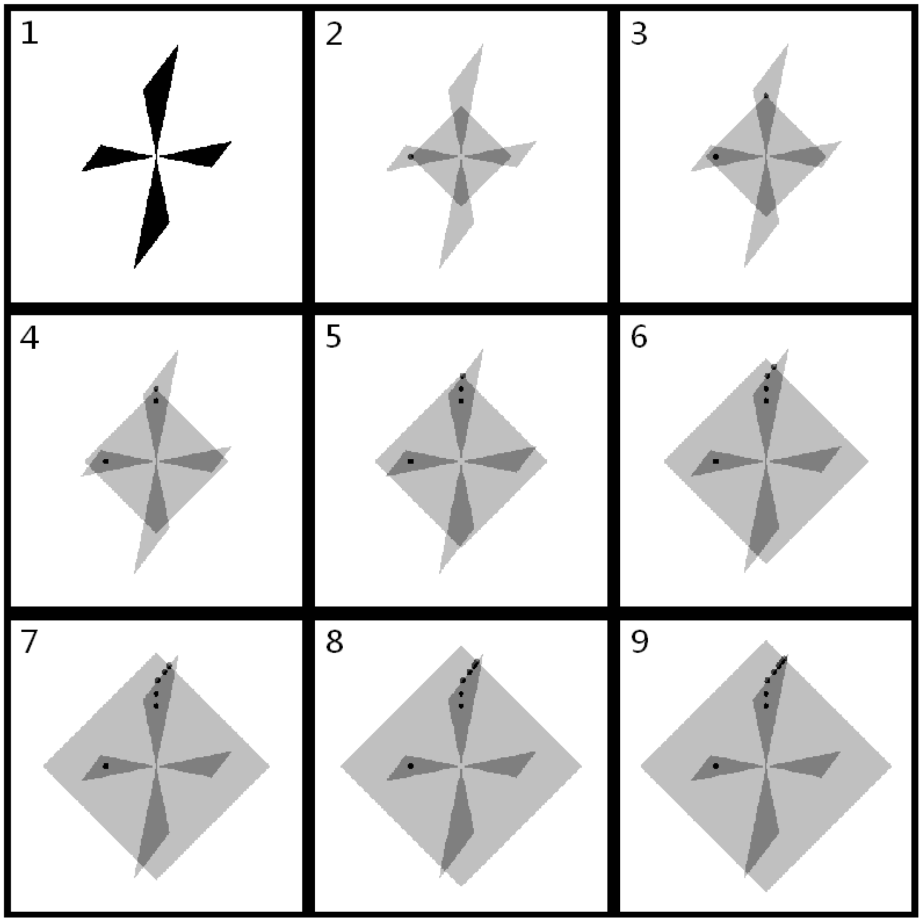


Figure 4.4: Illustration of naive fixed-point iteration (4.54) when $\diamond = \infty$.

$\|Q\boldsymbol{z}\|_\infty \leq 1, \|\boldsymbol{z}\|_1/\|\boldsymbol{z}\|_\infty \leq s\}$. Starting from an initial $\eta_0 = \|\boldsymbol{x}_0\|_\infty$ with $\boldsymbol{x}_0 \in \mathcal{S}$, the algorithm solves

$$\max_{\boldsymbol{z}} \|\boldsymbol{z}\|_\infty \text{ s.t. } \quad \|Q\boldsymbol{z}\|_\diamond \leq 1, \|\boldsymbol{z}\|_1 \leq s \cdot \eta_0 \tag{4.55}$$

in sub-figure 2. The solution is denoted by the black dot. Although the true domain for the optimization in (4.55) is the intersection of the distorted ℓ_∞ ball $\{\boldsymbol{z} : \|Q\boldsymbol{z}\|_\infty \leq 1\}$ and the ℓ_1 ball $\{\boldsymbol{z} : \|\boldsymbol{z}\|_1 \leq 1\}$, the intersection of the ℓ_1 ball (light gray diamond) and the star-shaped region $\mathcal{S}$ forms the effective domain, which is the dark grey region in the sub-figures. To see this, we note the optimal value of the optimization (4.55) $\eta_1 = \|\boldsymbol{x}_1\|_\infty \geq \eta_0$. As a consequence, for the optimal solution $\boldsymbol{x}_1$, we have $\|\boldsymbol{x}_1\|_1/\|\boldsymbol{x}_1\|_\infty \leq \|\boldsymbol{x}_1\|_1/\eta_0 \leq s$. In the following sub-figures, at each iteration, we expand the ℓ_1 ball until we get to the tip point of the star-shaped region $\mathcal{S}$, which is the global optimum.

Despite of its simplicity, the naive fixed-point iteration has two major disadvantages. Firstly, the stopping criterion based on successive improvement is not accurate as it does not reflect the gap between η_t and η^*. This disadvantage can be remedied by starting from both below and above η^*. The distance between corresponding terms in the two generated sequences is an indication of the gap to the fixed point η^*. However, the resulting algorithm is very slow, especially when updating η_{t+1} from above η^*. Secondly, the iteration process is slow, especially when close to the fixed point η^*. This is because $\rho_1(\epsilon)$ and $\rho_2(\epsilon)$ in 8) of Theorem 4.3.1 are close to 1.

2. **Bisection:** The bisection approach is motivated by property 7) of Theorem 4.3.1. Starting from an initial interval $(\eta_{\mathrm{L}}, \eta_{\mathrm{U}})$ that contains η^*, we compute $f_s(\eta_{\mathrm{M}})$ with $\eta_{\mathrm{M}} = (\eta_{\mathrm{L}} + \eta_{\mathrm{U}})/2$. As a consequence of property 7), $f_s(\eta_{\mathrm{M}}) > \eta_{\mathrm{M}}$ implies $f_s(\eta_{\mathrm{M}}) < \eta^*$, and we set $\eta_{\mathrm{L}} = f_s(\eta_{\mathrm{M}})$; $f_s(\eta_{\mathrm{M}}) < \eta_{\mathrm{M}}$ implies $f_s(\eta_{\mathrm{M}}) > \eta^*$, and we set $\eta_{\mathrm{U}} = f_s(\eta_{\mathrm{M}})$. The bisection process can also be accelerated by setting $\eta_{\mathrm{L}} = f_{s,i}(\eta_{\mathrm{M}})$ for the first $f_{s,i}(\eta_{\mathrm{M}})$ greater than η_{M}. The convergence of the bisection approach is much faster than the naive fixed point iteration because each iteration reduces the interval length at least by half. In addition, half the length of the interval is an upper bound on the gap between η_{M} and η^*, resulting an accurate stopping criterion. However, if the initial η_{U} is too larger than η^*, the majority of $f_s(\eta_{\mathrm{M}})$ would turn out to be less than η^*. The verification of $f_s(\eta_{\mathrm{M}}) < \eta_{\mathrm{M}}$ needs solving n linear programs or second-order cone programs, greatly degrading the algorithm's performance.

3. **Fixed-Point Iteration + Bisection:** The third approach combines the advantages of the bisection method and the fixed-point iteration method, at the level of $f_{s,i}(\eta)$. This method relies heavily on the representation $f_s(\eta) = \max_i f_{s,i}(\eta)$ and $\eta^* = \max_i \eta_i^*$.

   Starting from an initial interval $(\eta_{\mathrm{L}0}, \eta_{\mathrm{U}})$ and the index set $\mathcal{I}_0 = \{1, \ldots, n\}$, we pick any $i_0 \in \mathcal{I}_0$ and use the (accelerated) bisection method with starting interval $(\eta_{\mathrm{L}0}, \eta_{\mathrm{U}})$ to find the positive fixed point $\eta_{i_0}^*$ of $f_{s,i_0}(\eta)$. For any $i \in \mathcal{I}_0/i_0$, $f_{s,i}(\eta_{i_0}^*) \leq \eta_{i_0}^*$ implies that the fixed point η_i^* of $f_{s,i}(\eta)$ is less than or equal to $\eta_{i_0}^*$ according to the continuity of $f_{s,i}(\eta)$ and the uniqueness of its positive fixed point. As a consequence, we remove this i from the index set $\mathcal{I}_0$. We denote $\mathcal{I}_1$ as the index set after all such is is removed, *i.e.,* $\mathcal{I}_1 = \mathcal{I}_0/\{i : f_{s,i}(\eta^*_{i_0}) \leq \eta^*_{i_0}\}$. We also set $\eta_{L1} = \eta^*_{i_0}$ as $\eta^* \geq \eta^*_{i_0}$. Next we test the $i_1 \in \mathcal{I}_1$ with the *largest* $f_{s,i}(\eta^*_{i_0})$ and construct $\mathcal{I}_2$ and $\eta_{\text{L}2}$ in a similar manner. We repeat the process until the index set $\mathcal{I}_t$ is empty. The η^*_i found at the last step is the maximal η^*_i, which is equal to η^*.

Note that in equations (4.24), (4.25), and (4.27), if we replace the ℓ_∞ norm with any other norm (with some other minor modifications), especially $\|\cdot\|_{s,1}$ or $\|\cdot\|_2$, then a naive fixed-point iteration algorithm still exists. In addition, as we did in Corollary 4.1.1, we can express other norms on the error vector in terms of $\|\cdot\|_{s,1}$ and $\|\cdot\|_2$. We expect the norm $\|\cdot\|_{s,1}$ would yield the best performance bounds. Unfortunately, the major problem is that in these cases, the function $f_s(\eta)$ do not admit an obvious polynomial time algorithm to compute. It is very likely the corresponding norm maximization defining $f_s(\eta)$ for $\|\cdot\|_{s,1}$ and $\|\cdot\|_2$ are NP hard [84].

4.4 Probabilistic Analysis

In this section, we analyze the probabilistic behavior of $\omega_\diamond(Q, s)$ for random sensing matrices. More precisely, we will establish a result similar to (3.26) that guarantees $\omega_\diamond(Q, s)$ bounded away from zero with high probability.

For this purpose, we define the ℓ_1-constrained minimal singular value (ℓ_1-CMSV), which is connected with $\omega_\diamond(Q, s)$ and is more amenable to theoretical analysis:

Definition 4.4.1. *For any $s \in [1, n]$ and matrix $A \in \mathbb{R}^{m\times n}$, define the ℓ_1-constrained minimal singular value (abbreviated as ℓ_1-CMSV) of A by*

$$\rho_s(A) = \min_{\boldsymbol{z}:\ \|\boldsymbol{z}\|_1^2/\|\boldsymbol{z}\|_2^2 \leq s} \frac{\|A\boldsymbol{z}\|_2}{\|\boldsymbol{z}\|_2}. \tag{4.56}$$

Despite the seeming resemblance of the definitions for $\omega_\diamond(Q, s)$, especially $\omega_2(A, s)$, and $\rho_s(A)$, the difference in the ℓ_∞ norm and the ℓ_2 norm has important implications. As shown in Theorem 4.3.1, the ℓ_∞ norm enables the design of optimization procedures with nice convergence properties to efficiently compute $\omega_\diamond(Q, s)$. On the other hand, the ℓ_1-CMSV yields tight performance bounds at least for a large class of random sensing matrices, as we will see in Theorem 4.4.1.

However, there are some interesting connections among these quantities, as shown in the following proposition. These connections allow us the analyze the probabilistic behaviors of $\omega_\diamond(Q, s)$ using the results for $\rho_s(A)$ established in Theorem 4.4.1.

Proposition 4.4.1.

$$\sqrt{s}\sqrt{\omega_\infty(A^T A, s)} \geq \omega_2(A, s) \geq \rho_{s^2}(A). \tag{4.57}$$

Proof. For any $\boldsymbol{z}$ such that $\|\boldsymbol{z}\|_\infty = 1$ and $\|\boldsymbol{z}\|_1 \leq s$, we have

$$\begin{aligned} \boldsymbol{z} A^T A \boldsymbol{z} &\leq \sum_i |\boldsymbol{z}_i| |(A^T A \boldsymbol{z})_i| \\ &\leq \|\boldsymbol{z}\|_1 \|A^T A \boldsymbol{z}\|_\infty \\ &\leq s \|A^T A \boldsymbol{z}\|_\infty. \end{aligned} \tag{4.58}$$

Taking the minimum over $\{\boldsymbol{z} : \|\boldsymbol{z}\|_\infty = 1, \|\boldsymbol{z}\|_1 \leq s\}$ yields

$$\omega_2^2(A, s) \leq s\omega_\infty(A^T A, s). \tag{4.59}$$

Note that $\|\boldsymbol{z}\|_1/\|\boldsymbol{z}\|_\infty \leq s$ implies $\|\boldsymbol{z}\|_1 \leq s\|\boldsymbol{z}\|_\infty \leq s\|\boldsymbol{z}\|_2$, or equivalently,

$$\{\boldsymbol{z} : \|\boldsymbol{z}\|_1/\|\boldsymbol{z}\|_\infty \leq s\} \subseteqq \{\boldsymbol{z} : \|\boldsymbol{z}\|_1/\|\boldsymbol{z}\|_2 \leq s\}. \tag{4.60}$$

As a consequence, we have

$$\begin{aligned} \omega_2(A, s) &= \min_{\|\boldsymbol{z}\|_1/\|\boldsymbol{z}\|_\infty \leq s} \frac{\|A\boldsymbol{z}\|_2}{\|\boldsymbol{z}\|_2} \frac{\|\boldsymbol{z}\|_2}{\|\boldsymbol{z}\|_\infty} \\ &\geq \min_{\|\boldsymbol{z}\|_1/\|\boldsymbol{z}\|_\infty \leq s} \frac{\|A\boldsymbol{z}\|_2}{\|\boldsymbol{z}\|_2} \\ &\geq \min_{\|\boldsymbol{z}\|_1/\|\boldsymbol{z}\|_2 \leq s} \frac{\|A\boldsymbol{z}\|_2}{\|\boldsymbol{z}\|_2} \\ &= \rho_{s^2}(A), \end{aligned} \tag{4.61}$$

where the first inequality is due to $\|\boldsymbol{z}\|_2 \geq \|\boldsymbol{z}\|_\infty$, and the second inequality is because the minimization is taken over a larger set. $\square$

We first establish a probabilistic result on $\rho_s(A)$. In particular, we derive a condition on the number of measurements to get $\omega_\diamond(Q, s)$ bounded away from zero with high probability for sensing matrices with *i.i.d.* subgaussian and isotropic rows. Recall that a random vector $\boldsymbol{\xi} \in \mathbb{R}^n$ is called *isotropic and subgaussian* with constant L if $\mathbb{E}|\langle \boldsymbol{\xi}, \boldsymbol{u}\rangle|^2 = \|\boldsymbol{u}\|_2^2$ and $\mathbb{P}(|\langle \boldsymbol{\xi}, \boldsymbol{u}\rangle| \geq t) \leq 2\exp(-t^2/(L\|\boldsymbol{u}\|_2))$ hold for any $\boldsymbol{u} \in \mathbb{R}^n$.

Theorem 4.4.1. *Let the rows of the scaled sensing matrix $\sqrt{m}A$ be* i.i.d. *subgaussian and isotropic random vectors with numerical constant L. Then there exist constants c_1 and c_2 such that for any $\epsilon > 0$ and $m \geq 1$ satisfying*

$$m \geq c_1 \frac{L^2 s \log n}{\epsilon^2}, \tag{4.62}$$

we have

$$\mathbb{E}|1 - \rho_s(A)| \leq \epsilon, \tag{4.63}$$

and

$$\mathbb{P}\{1 - \epsilon \leq \rho_s(A) \leq 1 + \epsilon\} \geq 1 - \exp(-c_2\epsilon^2 m/L^4). \tag{4.64}$$

Proof. We connect the ℓ_1-CMSV for the sensing matrix A with an empirical process. Suppose the rows of $\sqrt{m}A$ are *i.i.d.* isotropic and subgaussian random vectors with constant L, and are denoted by $\{\boldsymbol{a}_i^T, i = 1, \ldots, m\}$. Denote $\mathcal{H}_s^n = \{\boldsymbol{u} \in \mathbb{R}^n : \|\boldsymbol{u}\|_2^2 = 1, \|\boldsymbol{u}\|_1^2 \leq s\}$, a subset of the unit sphere of $\mathbb{R}^n$. We observe that for any $\epsilon \in (0, 1)$

$$\rho_s^2(A) = \min_{\boldsymbol{u}:\boldsymbol{u}\in\mathcal{H}_s^n} \boldsymbol{u}^T A^T A \boldsymbol{u} < (1-\epsilon)^2 < (1-\epsilon) \tag{4.65}$$

is a consequence of

$$\begin{aligned} & \sup\nolimits_{\boldsymbol{u}\in\mathcal{H}_s^n} \left| \frac{1}{m}\boldsymbol{u}^T(\sqrt{m}A)^T(\sqrt{m}A)\boldsymbol{u} - 1 \right| \\ = \; & \sup\nolimits_{\boldsymbol{u}\in\mathcal{H}_s^n} \left| \frac{1}{m}\sum_{i=1}^m \langle \boldsymbol{a}_i, \boldsymbol{u}\rangle^2 - 1 \right| \leq \epsilon. \end{aligned} \tag{4.66}$$

Define a class of functions parameterized by $\boldsymbol{u}$ as $\mathcal{F}_s = \{f_{\boldsymbol{u}}(\cdot) = \langle \boldsymbol{u}, \cdot\rangle : \boldsymbol{u} \in \mathcal{H}_s^n\}$. Since $\mathbb{E}f^2(\boldsymbol{a}) = \mathbb{E}\langle \boldsymbol{u}, \boldsymbol{a}\rangle^2 = \|\boldsymbol{u}\|_2^2 = 1$ due to isotropy, the proof of Theorem 4.4.1 boils down to estimating

$$\mathbb{E}\ \sup_{f\in\mathcal{F}_s}\left|\frac{1}{m}\sum_{k=1}^{m} f^2(\boldsymbol{a}_i) - \mathbb{E}f^2(\boldsymbol{a})\right| \tag{4.67}$$

and

$$\mathbb{P}\left\{\sup_{f\in\mathcal{F}_s}\left|\frac{1}{m}\sum_{k=1}^{m} f^2(\boldsymbol{a}_i) - \mathbb{E}f^2\right|\right\} \tag{4.68}$$

using Theorem 2.3.2. The symmetry of $\mathcal{H}_s^n$ (and hence of $\mathcal{F}_s$) yields

$$\begin{aligned} \alpha &= \operatorname{diam}(\mathcal{F}_s, \|\cdot\|_{\psi_2}) \\ &= 2\ \sup_{\boldsymbol{u}\in\mathcal{H}_s^n}\|\langle \boldsymbol{u}, \boldsymbol{a}\rangle\|_{\psi_2} \le 2L. \end{aligned} \tag{4.69}$$

Now the key is to compute $\ell_*(\mathcal{H}_s^n)$ (actually an upper bound suffices). Clearly, we have

$$\begin{aligned} \ell_*(\mathcal{H}_s^n) &= \mathbb{E}\sup_{\boldsymbol{u}\in\mathcal{H}_s^n}\langle \boldsymbol{g}, \boldsymbol{u}\rangle \\ &\le \mathbb{E}\|\boldsymbol{u}\|_1\|\boldsymbol{g}\|_\infty \\ &\le \sqrt{s\log n}. \end{aligned} \tag{4.70}$$

The conclusions of Theorem 4.4.1 then follow from (6.134) and Theorem 2.3.2 with suitable choice of c_1. □

Following the proof procedure in [85] and using several novel entropy number estimates, we can also establish a result similar to (but slightly worse than) that of Theorem 4.4.1 for bounded orthonormal systems [86], which include the partial Fourier matrix and Hadamard matrix as important special cases.

Combining Proposition 4.4.1 and Theorem 4.4.1, we obtain the following probabilistic result on $\omega_\diamond(Q, s)$ for subgaussian and isotropic matrices.

Theorem 4.4.2. *Under the assumptions and notations of Theorem 4.4.1, there exist constants c_1 and c_2 such that for any $\epsilon > 0$ and $m \geq 1$ satisfying*

$$m \geq c_1 \frac{L^2 s^2 \log n}{\epsilon^2}, \tag{4.71}$$

we have

$$\mathbb{E}\ \omega_2(A, s) \geq 1 - \epsilon, \tag{4.72}$$

$$\mathbb{P}\{\omega_2(A, s) \geq 1 - \epsilon\} \geq 1 - \exp(-c_2 \epsilon^2 m), \tag{4.73}$$

and

$$\mathbb{E}\ \omega_\infty(A^T A, s) \geq \frac{(1-\epsilon)^2}{s}, \tag{4.74}$$

$$\mathbb{P}\left\{\omega_\infty(A, s) \geq \frac{(1-\epsilon)^2}{s}\right\} \geq 1 - \exp(-c_2 \epsilon^2 m). \tag{4.75}$$

As mentioned in Section 2.3.1, sensing matrices with *i.i.d.* subgaussian and isotropic rows include the Gaussian ensemble, and the Bernoulli ensemble, as well as the normalized volume measure on various convex symmetric bodies, for example, the unit balls of ℓ_p^n for $2 \leq p \leq \infty$ [51]. In equations (4.74) and (4.75), the extra s in the lower bound of $\omega_\infty(A^T A, s)$ would contribute an s factor in the bounds of Theorem 4.1.1. It plays the same role as the extra $\sqrt{k}$ factor in the error bounds for the DS and the LASSO in terms of the RIC and the ℓ_1-CMSV [69, 87].

The measurement bound (4.71) implies that the algorithms for verifying $\omega_\diamond > 0$ and for computing $\omega_\diamond$ work for s at least up to the order $\sqrt{m/\log n}$. The order $\sqrt{m/\log n}$ is complementary to the $\sqrt{m}$ upper bound in Proposition 4.2.1.

Note that Theorem 4.4.1 implies that the following program:

$$\max_{\boldsymbol{z}} \|\boldsymbol{z}\|_2 \text{ s.t. } A\boldsymbol{z} = 0, \|\boldsymbol{z}\|_1 \leq 1, \tag{4.76}$$

verifies the sufficient condition for exact ℓ_1 recovery for s up to the order $m/\log n$. Unfortunately, this program is NP hard and hence not tractable.

4.5 Numerical Simulations

In this section, we provide implementation details and numerically assess the performance of the algorithms for solving (4.19) and (4.24). All the numerical experiments in this section were conducted on a desktop computer with a Pentium D CPU@3.40GHz, 2GB RAM, and Windows XP operating system, and the computations were running single-core.

The n linear programs (4.21) for solving (4.19) are reformulated as the following standard linear programs:

$$\begin{aligned} &\min \begin{bmatrix} \mathbf{e}_i^T & \mathbf{0}^T \end{bmatrix} \begin{bmatrix} \boldsymbol{z} \\ \boldsymbol{u} \end{bmatrix} \\ &\text{s.t. } \begin{bmatrix} Q & \boldsymbol{O} \end{bmatrix} \begin{bmatrix} \boldsymbol{z} \\ \boldsymbol{u} \end{bmatrix} = \mathbf{0} \\ &\begin{bmatrix} \mathrm{I} & -\mathrm{I} \\ -\mathrm{I} & -\mathrm{I} \\ \mathbf{0}^T & \mathbf{1}^T \end{bmatrix} \begin{bmatrix} \boldsymbol{z} \\ \boldsymbol{u} \end{bmatrix} \leq \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ 1 \end{bmatrix} \end{aligned} \tag{4.77}$$

for $i = 1, \ldots, n$. These linear programs are implemented using the primal-dual algorithm outlined in Chapter 11 of [43]. The algorithm finds the optimal solution together with optimal dual vectors by solving the Karush-Kuhn-Tucker condition using linearization. The major computation load is solving linear systems of equations with positive definite coefficient matrices.

Recall that the optimization defining $f_{s,i}(\eta)$ is

$$\min \boldsymbol{z}_i \text{ s.t. } \|Q\boldsymbol{z}\|_\diamond \leq 1, \|\boldsymbol{z}\|_1 \leq s\eta. \tag{4.78}$$

Depending on whether $\diamond = 1, \infty$, or 2, (4.78) is solved using either linear programs or second-order cone programs. For example, when $\diamond = \infty$, we have the following corresponding linear programs:

$$\begin{aligned}&\min \begin{bmatrix} \mathbf{e}_i^T & \mathbf{0}^T \end{bmatrix} \begin{bmatrix} \boldsymbol{z} \\ \boldsymbol{u} \end{bmatrix} \\ &\text{s.t.} \begin{bmatrix} Q & \boldsymbol{O} \\ -Q & \boldsymbol{O} \\ \mathrm{I} & -\mathrm{I} \\ -\mathrm{I} & -\mathrm{I} \\ \mathbf{0}^T & \mathbf{1}^T \end{bmatrix} \begin{bmatrix} \boldsymbol{z} \\ \boldsymbol{u} \end{bmatrix} \leq \begin{bmatrix} \mathbf{1} \\ \mathbf{1} \\ \mathbf{0} \\ \mathbf{0} \\ s\eta \end{bmatrix}, i = 1, \dots, n,\end{aligned} \tag{4.79}$$

which are solved using the primal-dual algorithm similar to that for (4.77). When $\diamond = 2$, we rewrite (4.78) as the following second-order cone programs

$$\begin{aligned}&\min \begin{bmatrix} \mathbf{e}_i^T & \mathbf{0}^T \end{bmatrix} \begin{bmatrix} \boldsymbol{z} \\ \boldsymbol{u} \end{bmatrix} \\ &\text{s.t. } \frac{1}{2}\left(\left\| \begin{bmatrix} Q & \boldsymbol{O} \end{bmatrix} \begin{bmatrix} \boldsymbol{z} \\ \boldsymbol{u} \end{bmatrix} \right\|_2^2 - 1 \right) \leq 0 \\ &\begin{bmatrix} \mathrm{I} & -\mathrm{I} \\ -\mathrm{I} & -\mathrm{I} \\ \mathbf{0}^T & \mathbf{1}^T \end{bmatrix} \begin{bmatrix} \boldsymbol{z} \\ \boldsymbol{u} \end{bmatrix} \leq \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ s\eta \end{bmatrix}.\end{aligned} \tag{4.80}$$

We use the log-barrier algorithm described in Chapter 11 of [43] to solve (4.80). Interested readers are encouraged to refer to [79] for a concise exposition of the general primal-dual and log-barrier algorithms and implementation details for similar linear programs and second-order cone programs.

We test the algorithms on Bernoulli, Gaussian, and Hadamard matrices of different sizes. The entries of Bernoulli and Gaussian matrices are randomly generated from the classical Bernoulli distribution with equal probability and the standard Gaussian distribution, respectively. For Hadamard matrices, first a square Hadamard matrix of size n (n is a power of 2) is generated, then its rows are randomly permuted and its first m rows are taken as an $m \times n$ sensing matrix. All $m \times n$ matrices are normalized to have columns of unit length.

The performance of the verification program (4.20) was previously reported in [87] in another context. For completeness, we reproduce some of the results in Tables 4.1, 4.2, and 4.3. We present the calculated lower bounds for k^*, defined as the maximal sparsity level such that the sufficient and necessary condition (3.17) holds. We compare the lower bounds and the running times for our implementation of (4.19), denoted as L_∞, and for the algorithm given in [40], denoted as JN (the authors' initials). Apparently, the lower bound on k^* computed by L_∞ is $k_* \overset{\text{def}}{=} \lfloor s_*/2 \rfloor$ with s_* given by (4.19). Table 4.1 and 4.2 are for Hadamard and Gaussian matrices, respectively, with $n = 256$ and $m = \lfloor \rho n \rfloor, \rho = 0.1, 0.2, \ldots, 0.9$. Table 4.3 is for matrices with a leading dimension $n = 1024$. We conclude that L_∞ and JN give comparable lower bounds, and our implementation of L_∞ performs much faster. In addition, it consumes less memory and is very stable, as seen by the variation in execution times.

Table 4.1: Comparison of L_∞ and JN for a Hadamard matrix with leading dimension $n = 256$.

m	lower bound on k^*		CPU time (s)	
	L_∞	JN	L_∞	JN
25	1	1	3	35
51	2	2	6	70
76	3	3	7	102
102	4	4	9	303
128	5	5	9	544
153	7	7	13	310
179	9	9	15	528
204	12	12	18	1333
230	19	18	18	435

In the next set of experiments, we assess the performance of (4.79) and (4.80) for computing $\omega_\infty(A^T A, s)$ and $\omega_2(A, s)$, respectively. We compare our recovery error bounds based on $\omega_\diamond$ with those based on the RIC. Combining Corollary 4.1.1 and Theorem 4.1.1, we have for the Basis Pursuit

$$\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2 \leq \frac{2\sqrt{2k}}{\omega_2(A, 2k)} \epsilon, \tag{4.81}$$

Table 4.2: Comparison of L_∞ and JN for a Gaussian matrix with leading dimension $n = 256$.

m	lower bound on k^*		CPU time (s)	
	L_∞	JN	L_∞	JN
25	1	1	6	91
51	2	2	8	191
76	3	3	10	856
102	4	4	13	5630
128	4	5	16	5711
153	6	6	20	1381
179	7	7	24	3356
204	10	10	25	10039
230	13	14	28	8332

Table 4.3: Comparison of L_∞ and JN for Gaussian and Hadamard matrices with leading dimension $n = 1024$. In the column head, "G" represents Gaussian matrix and "H" represents Hadamard matrix.

m	lower bound on k^*			CPU time (s)		
	L_∞(H)	L_∞(G)	JN(G)	L_∞(H)	L_∞(G)	JN(G)
102	3	2	2	182	136	457
204	4	4	4	501	281	1179
307	6	6	6	872	510	2235
409	8	7	7	1413	793	3659
512	11	10	10	1914	990	5348
614	14	12	12	1362	1309	7156
716	18	15	15	1687	1679	9446
819	24	20	21	1972	2033	12435
921	37	29	32	2307	2312	13564

and for the Dantzig selector

$$\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2 \leq \frac{2\sqrt{2k}}{\omega_\infty(A^T A, 2k)}\mu. \tag{4.82}$$

For comparison, the two RIC bounds are

$$\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2 \leq \frac{4\sqrt{1+\delta_{2k}(A)}}{1-(1+\sqrt{2})\delta_{2k}(A)}\epsilon, \tag{4.83}$$

for the Basis Pursuit, assuming $\delta_{2k}(A) < \sqrt{2}-1$ [39], and

$$\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2 \leq \frac{4\sqrt{k}}{1-\delta_{2k}(A)-\delta_{3k}(A)}\mu, \tag{4.84}$$

for the Dantzig selector, assuming $\delta_{2k}(A)+\delta_{3k}(A) < 1$ [69]. Without loss of generality, we set $\epsilon = 1$ and $\mu = 1$.

The RIC is computed using Monte Carlo simulations. More explicitly, for $\delta_{2k}(A)$, we randomly take 1000 sub-matrices of $A \in \mathbb{R}^{m\times n}$ of size $m \times 2k$, compute the maximal and minimal singular values σ_1 and σ_{2k}, and approximate $\delta_{2k}(A)$ using the maximum of $\max(\sigma_1^2 - 1, 1 - \sigma_{2k}^2)$ among all sampled sub-matrices. Obviously, the approximated RIC is always smaller than or equal to the exact RIC. As a consequence, the performance bounds based on the exact RIC are *worse* than those based on the approximated RIC. Therefore, in cases where our $\omega_\diamond$ based bounds are better (tighter, smaller) than the approximated RIC bounds, they are even better than the exact RIC bounds.

In Tables 4.4, 4.5, and 4.6, we compare the error bounds (4.81) and (4.83) for the Basis Pursuit algorithm. The corresponding s_* and k_* for different m are also included in the tables. Note the blanks mean that the corresponding bounds are not valid. For the Bernoulli and Gaussian matrices, the RIC bounds work only for $k \leq 2$, even with $m = \lfloor 0.8n \rfloor$, while the $\omega_2(A, 2k)$ bounds work up until $k = 9$. Both bounds are better for Hadamard matrices. For example, when $m = 0.5n$, the RIC bounds are valid for $k \leq 3$, and our bounds hold for $k \leq 5$. In all cases for $n = 256$, our bounds are smaller than the RIC bounds.

We next compare the error bounds (4.82) and (4.84) for the Dantzig selector. For the Bernoulli and Gaussian matrices, our bounds work for wider ranges of (k, m) pairs and are tighter in all test cases. For the Hadamard matrices, the RIC bounds are better, starting from $k \geq 5$ or 6. We expect that this indicates a general trend, namely, when k is relatively small, the ω based bounds are better, while when k is

Table 4.4: Comparison of the ω_2 based bounds and the RIC based bounds on the ℓ_2 norms of the errors of the Basis Pursuit algorithm for a Bernoulli matrix with leading dimension $n = 256$.

	m	51	77	102	128	154	179	205
	s_*	4.6	6.1	7.4	9.6	12.1	15.2	19.3
k	k_*	2	3	3	4	6	7	9
1	ω bd	4.2	3.8	3.5	3.4	3.3	3.2	3.2
	ric bd			23.7	16.1	13.2	10.6	11.9
2	ω bd	31.4	12.2	9.0	7.4	6.5	6.0	5.6
	ric bd						72.1	192.2
3	ω bd		252.0	30.9	16.8	12.0	10.1	8.9
	ric bd							
4	ω bd				52.3	23.4	16.5	13.6
	ric bd							
5	ω bd					57.0	28.6	20.1
	ric bd							
6	ω bd					1256.6	53.6	30.8
	ric bd							
7	ω bd						161.6	50.6
	ric bd							
8	ω bd							93.1
	ric bd							
9	ω bd							258.7
	ric bd							

large, the RIC bounds are tighter. This was suggested by the probabilistic analysis of ω in Section 4.4. The reason is that when k is relatively small, both the relaxation $\|\boldsymbol{x}\|_1 \leq 2k\|\boldsymbol{x}\|_\infty$ on the sufficient and necessary condition (3.17) and the relaxation $\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2 \leq \sqrt{2k}\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_\infty$ are sufficiently tight.

In Table 4.10 we present the execution times for computing different ω. For random matrices with leading dimension $n = 256$, the algorithm generally takes 1 to 3 minutes to compute either $\omega_2(A, s)$ or $\omega_\infty(A^T A, s)$.

In the last set of experiments, we compute $\omega_2(A, 2k)$ and $\omega_\infty(A^T A, 2k)$ for a Gaussian matrix and a Hadamard matrix, respectively, with leading dimension $n = 512$. The row dimensions of the sensing matrices range over $m = \lfloor \rho n \rfloor$ with $\rho = 0.2, 0.3, \ldots, 0.8$.

Table 4.5: Comparison of the ω_2 based bounds and the RIC based bounds on the ℓ_2 norms of the errors of the Basis Pursuit algorithm for a Hadamard matrix with leading dimension $n = 256$.

	m	51	77	102	128	154	179	205
	s_*	5.4	7.1	9.1	11.4	14.0	18.4	25.3
k	k_*	2	3	4	5	6	9	12
1	ω bd	3.8	3.5	3.3	3.2	3.1	3.0	3.0
	ric bd	46.6	13.2	9.2	9.4	8.3	6.2	5.2
2	ω bd	13.7	8.4	6.7	5.9	5.4	4.9	4.6
	ric bd			46.6	24.2	15.3	8.6	7.1
3	ω bd		30.9	14.0	10.1	8.4	7.1	6.3
	ric bd				1356.6	25.4	10.3	8.8
4	ω bd			47.4	18.9	13.2	9.9	8.1
	ric bd					40.0	14.0	10.2
5	ω bd				51.5	22.6	13.8	10.3
	ric bd						18.8	11.6
6	ω bd					50.8	20.1	13.1
	ric bd						42.5	15.9
7	ω bd						31.8	16.7
	ric bd						94.2	19.7
8	ω bd						63.5	21.7
	ric bd						1000.0	24.6
9	ω bd						449.8	29.4
	ric bd							39.1
10	ω bd							42.8
	ric bd							35.6
11	ω bd							72.7
	ric bd							134.1
12	ω bd							195.1
	ric bd							

In Figure 4.5, the quantities $\omega_2(A, 2k)$ and $\omega_\infty(A^T A, 2k)$ are plotted as a function of k for different m. We can clearly see how increasing the number of measurements (namely, the value of m) increases the goodness measure, and how increasing the sparsity level decreases the goodness measure.

In Figure 4.6, we compare the ℓ_2 norm error bounds of the Basis Pursuit using $\omega_2(A, 2k)$ and the RIC. The color indicates the values of the error bounds. We remove

Table 4.6: Comparison of the ω_2 based bounds and the RIC based bounds on the ℓ_2 norms of the errors of the Basis Pursuit algorithm for a Gaussian matrix with leading dimension $n = 256$.

	m	51	77	102	128	154	179	205
	s_*	4.6	6.2	8.1	9.9	12.5	15.6	20.0
k	k_*	2	3	4	4	6	7	10
1	ω bd	4.3	3.7	3.5	3.4	3.3	3.2	3.2
	ric bd			26.0	14.2	10.0	10.9	12.1
2	ω bd	34.3	12.3	8.3	7.0	6.4	5.9	5.6
	ric bd						47.1	27.6
3	ω bd		197.4	23.4	14.5	11.6	9.8	8.9
	ric bd							
4	ω bd			1036.6	39.6	21.7	15.9	13.4
	ric bd							
5	ω bd					49.3	26.4	20.0
	ric bd							
6	ω bd					284.2	48.8	31.2
	ric bd							
7	ω bd						129.1	48.1
	ric bd							
8	ω bd							185.5
	ric bd							
9	ω bd							9640.3
	ric bd							

all bounds that are greater than 50 or are not valid. Hence, all white areas indicate that the bounds corresponding to (k, m) pairs that are too large or not valid. The left sub-figure is based on $\omega_2(A, 2k)$ and the right sub-figure is based on the RIC. We observe that the $\omega_2(A, 2k)$ based bounds apply to a wider range of (k, m) pairs.

In Figure 4.7, we conduct the same experiment as in Figure 4.6 for a Hadamard matrix and the Dantzig selector. We observe that for the Hadamard matrix, the RIC gives better performance bounds. This result coincides with the one we obtained in Table 4.8.

The average time for computing each $\omega_2(A, 2k)$ and $\omega_\infty(A^T A, 2k)$ was around 15 minutes.

Table 4.7: Comparison of the ω_∞ based bounds and the RIC based bounds on the ℓ_2 norms of the errors of the Dantzig selector algorithm for the Bernoulli matrix used in Table 4.4.

	m	51	77	102	128	154	179	205
	s_*	4.6	6.1	7.4	9.6	12.1	15.2	19.3
k	k_*	2	3	3	4	6	7	9
1	ω bd	6.0	5.4	4.8	4.4	4.2	4.1	4.1
	ric bd		46.3	17.4	12.1	11.2	10.3	8.6
2	ω bd	102.8	38.4	29.0	18.5	14.1	12.8	11.9
	ric bd						47.2	22.5
3	ω bd		1477.2	170.2	81.2	57.0	41.1	32.6
	ric bd							
4	ω bd				522.7	194.6	128.9	89.0
	ric bd							
5	ω bd					768.7	323.6	203.2
	ric bd							
6	ω bd					24974.0	888.7	489.0
	ric bd							
7	ω bd						3417.3	1006.9
	ric bd							
8	ω bd							2740.0
	ric bd							
9	ω bd							10196.9
	ric bd							

Table 4.8: Comparison of the ω_∞ based bounds and the RIC based bounds on the ℓ_2 norms of the errors of the Dantzig selector algorithm for the Hadamard matrix used in Table 4.5.

	m	51	77	102	128	154	179	205
	s_*	5.2	6.9	9.1	12.1	14.4	18.3	25.2
k	k_*	2	3	4	6	7	9	12
1	ω bd	4.8	4.0	3.8	3.4	3.4	3.2	3.1
	ric bd		15.6	9.3	7.0	6.3	5.8	5.1
2	ω bd	50.9	16.2	10.1	7.1	7.0	6.1	5.3
	ric bd			45.3	16.6	13.7	10.6	8.8
3	ω bd		108.2	30.7	14.3	13.9	10.0	8.0
	ric bd			1016.4	29.9	24.9	15.8	12.5
4	ω bd			150.7	35.3	29.3	16.8	11.7
	ric bd				126.4	38.7	24.2	16.6
5	ω bd				108.5	64.2	31.4	17.3
	ric bd					187.3	30.0	22.1
6	ω bd				3168.9	171.5	59.7	25.3
	ric bd					112.0	53.1	26.8
7	ω bd					1499.5	116.3	38.8
	ric bd					411.7	71.3	34.7
8	ω bd						265.3	61.4
	ric bd						95.4	47.6
9	ω bd						2394.0	96.0
	ric bd						198.7	61.9
10	ω bd							157.4
	ric bd							82.9
11	ω bd							296.4
	ric bd							130.3
12	ω bd							898.2
	ric bd							201.2

Table 4.9: Comparison of the ω_∞ based bounds and the RIC based bounds on the ℓ_2 norms of the errors of the Dantzig selector algorithm for the Gaussian matrix used in Table 4.6.

	m	51	77	102	128	154	179	205
	s_*	4.6	6.2	8.1	9.9	12.5	15.6	20.0
k	k_*	2	3	4	4	6	7	10
1	ω bd	6.5	5.1	4.8	4.3	4.2	4.0	3.9
	ric bd		30.0	18.0	14.6	9.7	9.3	9.1
2	ω bd	119.4	37.8	22.5	17.6	14.1	12.7	11.4
	ric bd					91.5	44.4	23.5
3	ω bd		1216.7	120.7	67.3	53.6	38.7	36.4
	ric bd							2546.6
4	ω bd			4515.9	318.2	168.4	115.8	109.0
	ric bd							
5	ω bd					663.6	292.4	247.8
	ric bd							
6	ω bd					5231.4	764.3	453.5
	ric bd							
7	ω bd						2646.4	1087.7
	ric bd							
8	ω bd							2450.5
	ric bd							
9	ω bd							6759.0
	ric bd							

Table 4.10: Time in seconds taken to compute $\omega_2(A, \cdot)$ and $\omega_\infty(A^T A, \cdot)$ for Bernoulli, Hadamard, and Gaussian matrices

k	type	m	51	77	102	128	154	179	205
1	Bernoulli	ω_2	118	84	133	87	133	174	128
		ω_∞	75	81	84	65	63	144	151
	Hadamard	ω_2	84	82	82	82	80	79	79
		ω_∞	57	55	58	58	58	58	57
	Gaussian	ω_2	82	84	212	106	156	185	104
		ω_∞	69	65	72	102	81	104	72
3	Bernoulli	ω_2		155	96	95	97	97	131
		ω_∞		300	228	190	125	135	196
	Hadamard	ω_2		91	88	87	88	74	72
		ω_∞		84	83	77	92	102	70
	Gaussian	ω_2		134	168	115	95	96	100
		ω_∞		137	142	125	165	145	105
5	Bernoulli	ω_2					97	111	97
		ω_∞					156	81	107
	Hadamard	ω_2				87	85	85	81
		ω_∞				75	74	75	75
	Gaussian	ω_2					98	105	96
		ω_∞							193
7	Bernoulli	ω_2						164	104
		ω_∞						178	85
	Hadamard	ω_2						82	77
		ω_∞					134	71	65
	Gaussian	ω_2						106	105
		ω_∞							193

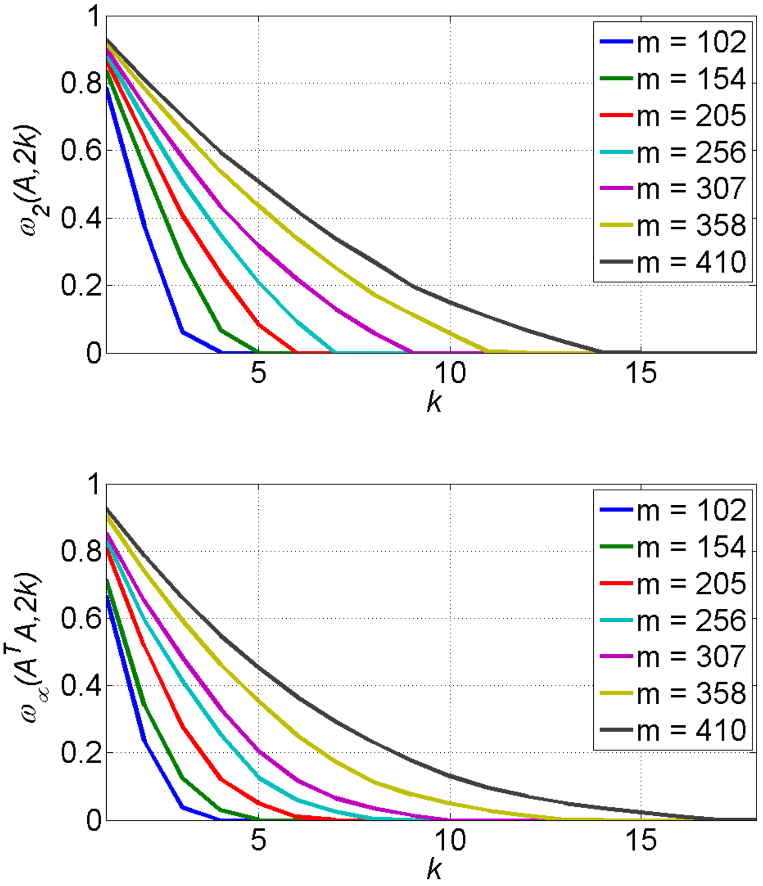


Figure 4.5: $\omega_2(A, 2k)$ and $\omega_\infty(A^T A, 2k)$ as a function of k with $n = 512$ and different ms. The top plot is for a Gaussian matrix, and the bottom plot is for a Hadamard matrix.

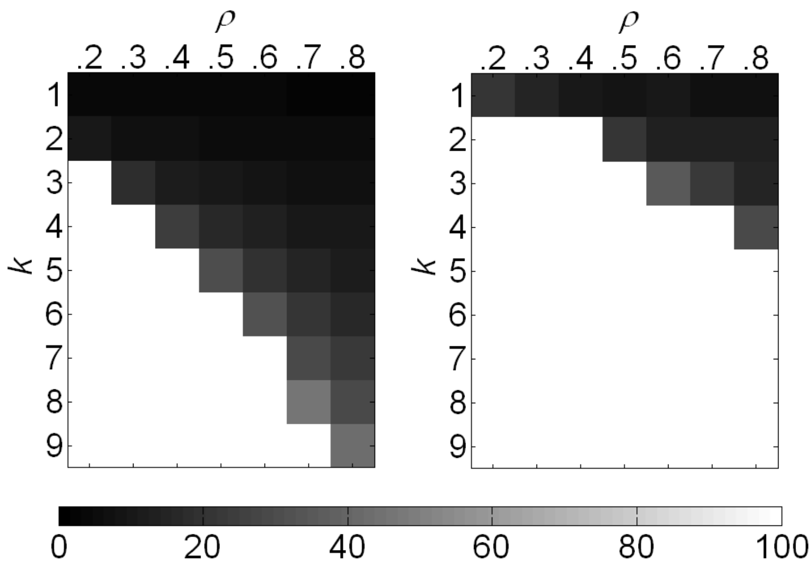


Figure 4.6: $\omega_2(A, 2k)$ based bounds v.s. RIC based bounds on the ℓ_2 norms of the errors for a Gaussian matrix with leading dimension $n = 512$. Left: $\omega_2(A, 2k)$ based bounds; Right: RIC based bounds.

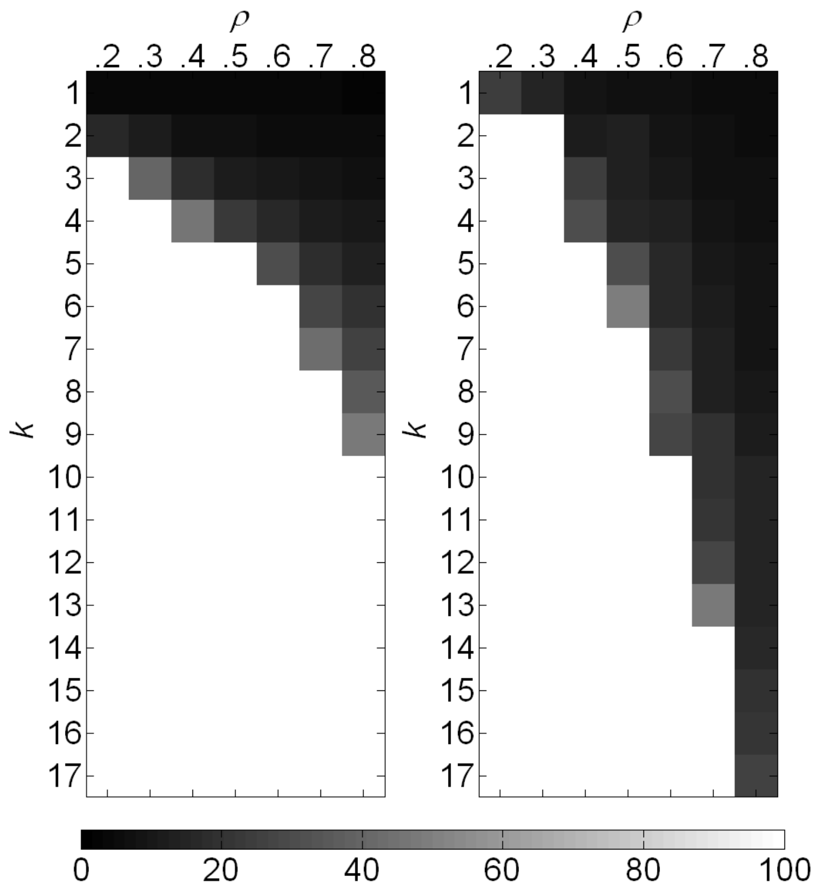


Figure 4.7: $\omega_\infty(A^TA, 2k)$ based bounds v.s. RIC based bounds on the ℓ_2 norms of the errors for a Hadamard matrix with leading dimension $n = 512$. Left: $\omega_2(A, 2k)$ based bounds; Right: RIC based bounds

Chapter 5

Block Sparsity Recovery: Background

In this chapter, we present background knowledge on block-sparsity recovery in a structure similar to Chapter 3.

5.1 Introduction to Block Sparsity Recovery

Block-sparsity is yet another important low-dimensional structure that can be exploited effectively in signal recovery. Mathematically, the recovery of signals with block-sparsity structures reconstructs a block-sparse signal from usually noisy linear measurements:

$$\boldsymbol{y} = A\boldsymbol{x} + \boldsymbol{w}, \tag{5.1}$$

where $\boldsymbol{x} \in \mathbb{R}^{np}$ is a signal with p blocks, each of length n, $A \in \mathbb{R}^{m\times np}$ is the measurement or sensing matrix, $\boldsymbol{y} \in \mathbb{R}^m$ is the measurement vector, and $\boldsymbol{w} \in \mathbb{R}^m$ is the noise vector. We assume $\boldsymbol{x}$ is a $k-$block-sparse signal, namely, the block-sparsity level of $\boldsymbol{x}$ is $\|\boldsymbol{x}\|_{\mathrm{b}0} = k \ll p$. The number of measurements m is much smaller than the signal dimension np, making the measurement system underdetermined.

Block-sparsity arises naturally in applications such as sensor arrays [12], radar [22], multi-band signals [88], and DNA microarrays [64]. A particular area that motivates this work is the application of block-sparse signal recovery in radar systems. The signals in radar applications are usually sparse because there are only a few targets to be estimated among many possibilities. However, a single target manifests itself simultaneously in the sensor domain, the frequency domain, the temporal domain, and the reflection-path domain. As a consequence, the underlying signal would be block-sparse if the radar system observes the targets from several of these domains [21, 22].

A very special block-sparse model is when $\boldsymbol{x}$ is a complex signal, such as the models in sensor array and radar applications. Note that the computable performance analysis developed in Chapter 4 for the real variables does not apply to the complex case.

5.2 Recovery Algorithms

Many algorithms in sparse signal recovery have been extended to recover the block-sparse signal $\boldsymbol{x}$ from $\boldsymbol{y}$ by exploiting the block-sparsity of $\boldsymbol{x}$. We focus on three algorithms based on block-ℓ_1 minimization: the Block-Sparse Basis Pursuit (BS-BP) [6], the Block-Sparse Dantzig selector (BS-DS) [89], and the Block-Sparse LASSO estimator (BS-LASSO) [90].

$$\text{BS-BP: } \min_{\boldsymbol{z}\in\mathbb{R}^{np}} \|\boldsymbol{z}\|_{\text{b}1} \ \text{ s.t. } \|\boldsymbol{y} - A\boldsymbol{z}\|_2 \leq \varepsilon \tag{5.2}$$

$$\text{BS-DS: } \min_{\boldsymbol{z}\in\mathbb{R}^{np}} \|\boldsymbol{z}\|_{\text{b}1} \ \text{ s.t. } \ \|A^T(\boldsymbol{y} - A\boldsymbol{z})\|_{\text{b}\infty} \leq \mu \tag{5.3}$$

$$\text{BS-LASSO: } \min_{\boldsymbol{z}\in\mathbb{R}^{np}} \frac{1}{2}\|\boldsymbol{y} - A\boldsymbol{z}\|_2^2 + \mu\|\boldsymbol{z}\|_{\text{b}1}. \tag{5.4}$$

Here μ is a tuning parameter, and ε is a measure of the noise level. All three optimization problems have efficient implementations using convex programming. For example, the BS-BP can be solved by the following second-order cone program:

$$\min_{\boldsymbol{z},\boldsymbol{u}} \mathbf{1}_p^T\boldsymbol{u} \ \text{ s.t. } \ \|A\boldsymbol{z} - \boldsymbol{y}\|_2 \leq \varepsilon, \|\boldsymbol{z}_{[i]}\|_2 \leq u_i, i = 1, \ldots, p. \tag{5.5}$$

In the noise-free case where $\boldsymbol{w} = 0$, roughly speaking all the three algorithms reduce to

$$\min_{\boldsymbol{z}\in\mathbb{R}^{np}} \|\boldsymbol{z}\|_{\text{b}1} \ \text{ s.t. } A\boldsymbol{z} = A\boldsymbol{x}, \tag{5.6}$$

which is the block-ℓ_1 relaxation of the block-ℓ_0 problem:

$$\min_{\boldsymbol{z}\in\mathbb{R}^n} \|\boldsymbol{z}\|_{\mathrm{b}0} \ \text{ s.t. } A\boldsymbol{z} = A\boldsymbol{x}. \tag{5.7}$$

5.3 Null Space Characterization and Restricted Isometry Property

A minimal requirement on the block-ℓ_1 minimization algorithms is the *uniqueness and exactness* of the solution $\hat{\boldsymbol{x}} \overset{\text{def}}{=} \operatorname{argmin}_{\boldsymbol{z}:A\boldsymbol{z}=A\boldsymbol{x}}\|\boldsymbol{x}\|_{\mathrm{b}1}$, *i.e.*, $\hat{\boldsymbol{x}} = \boldsymbol{x}$. When the true signal $\boldsymbol{x}$ is $k-$block-sparse, the sufficient and necessary condition for exact block-ℓ_1 recovery is given by the block-sparse version of NSP [37]

$$\sum_{i\in S} \|\boldsymbol{z}_{[i]}\|_2 < \sum_{i\notin S} \|\boldsymbol{z}_{[i]}\|_2, \forall \boldsymbol{z} \in \operatorname{null}(A), |S| \leq k, \tag{5.8}$$

or equivalently

$$\|\boldsymbol{z}_{[S]}\|_{\mathrm{b}1} < \frac{1}{2}\|\boldsymbol{z}\|_{\mathrm{b}1}, \forall \boldsymbol{z} \in \operatorname{null}(A), |S| \leq k. \tag{5.9}$$

Similar to the sparse version of NSP, the block-sparse version of NSP can be viewed as a block-ℓ_1 relaxation of the sufficient and necessary condition for unique recovery of (5.7). The unique and exact block-ℓ_0 recovery according to (5.7) is that for any index set $S \subset \{1, \ldots, p\}$ of size $2k$, the columns of $A_{[S]}$ are linearly independent, which can be equivalently expressed as

$$k < \frac{1}{2}\|\boldsymbol{z}\|_{\mathrm{b}0}, \forall \boldsymbol{z} \in \operatorname{null}(A), \tag{5.10}$$

or

$$\|\boldsymbol{z}_{[S]}\|_{\mathrm{b}0} < \|\boldsymbol{z}_{[S^c]}\|_{\mathrm{b}0}, \forall \boldsymbol{z} \in \operatorname{null}(A), \ |S| \leq k. \tag{5.11}$$

In the noisy case, extending the RIC to block-sparse vectors, we define the block restricted isometry constant (bRIC) as follows:

Definition 5.3.1. *Let $A \in \mathbb{R}^{m\times np}$ be a given matrix. For each integer $k \in \{1, \ldots, p\}$, the block restricted isometry constant (bRIC) $\delta_{\mathrm{b}k}$ is defined as the smallest $\delta > 0$ such that*

$$1 - \delta \leq \frac{\|A\boldsymbol{x}\|_2^2}{\|\boldsymbol{x}\|_2^2} \leq 1 + \delta \tag{5.12}$$

holds for arbitrary non-zero $k-$block-sparse signal $\boldsymbol{x}$.

A matrix A with a small $\delta_{\mathrm{b}k}$ is nearly an isometry when restricted onto all $k-$block-sparse vectors. Apparently, $\delta_{2k}(A) < 1$ if and only if for any $S \subset \{1, \ldots, p\}$ of size $2k$, the columns of the matrix $A_{[S]}$ are linearly independent. Therefore, $\delta_{\mathrm{b}2k}(A) < 1$ is the necessary and sufficient condition for exact block-ℓ_0 recovery. It is shown in [6] that $\delta_{\mathrm{b}2k}(A) < \sqrt{2} - 1$ is a sufficient condition for the unique and exact recovery of the block-ℓ_1 minimization algorithm (5.6), suggesting $\delta_{\mathrm{b}2k}(A) < \sqrt{2} - 1$ implies the block-sparse version of the NSP. The converse is not true.

Now we present some of performance bounds on the BP, the DS, and the LASSO in terms of the bRIC. Assume $\boldsymbol{x}$ is a $k-$block-sparse signal and $\hat{\boldsymbol{x}}$ is its estimate given by any of the three algorithms; then we have the following:

1. BS-BP [6]: *Suppose that $\delta_{\mathrm{b}2k} < \sqrt{2} - 1$ and $\|\boldsymbol{w}\|_2 \leq \varepsilon$. The solution to the BP* (3.11) *satisfies*

$$\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2 \leq \frac{4\sqrt{1 + \delta_{\mathrm{b}2k}}}{1 - (1 + \sqrt{2})\delta_{\mathrm{b}2k}} \cdot \varepsilon. \tag{5.13}$$

2. BS-DS: *If the noise $\boldsymbol{w}$ satisfies $\|A^T\boldsymbol{w}\|_\infty < \mu$, and $\delta_{\mathrm{b}2k} + \delta_{\mathrm{b}3k} < 1$, then, the error signal obeys*

$$\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2 \leq \frac{4\sqrt{k}}{1 - \delta_{\mathrm{b}2k} - \delta_{\mathrm{b}3k}}\mu. \tag{5.14}$$

3. BS-LASSO: *If the noise* $\boldsymbol{w}$ *satisfies* $\|A^T\boldsymbol{w}\|_\infty < \mu$, *and* $\delta_{\mathrm{b}2k} < 1/(3\sqrt{2}+1)$, *then, the error signal of* (3.13) *satisfies*

$$\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2 \quad \leq \quad \frac{16\sqrt{k}}{(1-\delta_{\mathrm{b}2k})\left(1 - \frac{3\sqrt{2}\delta_{\mathrm{b}2k}}{1-\delta_{\mathrm{b}2k}}\right)^2}\mu. \tag{5.15}$$

These performance bounds are exactly parallel to those on their sparse counterparts and the derivations are similar. However, we are not able to locate references for (5.14) and (5.15).

5.4 Probabilistic Analysis

Section 5.3 states that a sufficiently small bRIC guarantees stable recovery of a block-sparse signal from linear measurements. Similar to standard RIC, it has been shown that for a class of random sensing matrices, the bRIC δ_{b2k} is small with high probability. We present such a result below [6]:

Let $A \in \mathbb{R}^{m\times np}$ *be a random matrix whose entries are* i.i.d. *samples from the Gaussian distribution* $\mathcal{N}(0, 1/m)$. *Suppose* $t > 0$ *and* $0 < \delta < 1$ *are numerical constants. If*

$$m \geq \frac{36}{7\delta}\left(\log\left(2\binom{p}{k}\right) + kn\log\left(\frac{12}{\delta}\right) + t\right), \tag{5.16}$$

then $\delta_{\mathrm{b}2k}(A) < \delta$ *with probability at least* $1 - \exp(-t)$.

Since we have

$$(p/k)^k \leq \binom{p}{k} \leq (ep/k)^k, \tag{5.17}$$

the right hand side of (5.16) is roughly

$$c(\delta)\left(k\log\frac{p}{k} + kn\right). \tag{5.18}$$

Chapter 6

Computable Performance Analysis for Block-Sparsity Recovery

6.1 Goodness Measures and Error Bounds

In this section, we derive performance bounds on the block-ℓ_∞ norms of the error vectors. We first establish a proposition characterizing the error vectors of the block-ℓ_1 recovery algorithms.

Proposition 6.1.1. *Suppose* $\boldsymbol{x} \in \mathbb{R}^{np}$ *in* (5.1) *is* $k-$*block-sparse and the noise* $\boldsymbol{w}$ *satisfies* $\|\boldsymbol{w}\|_2 \leq \varepsilon$, $\|A^T\boldsymbol{w}\|_{\mathrm{b}\infty} \leq \mu$, *and* $\|A^T\boldsymbol{w}\|_\infty \leq \kappa\mu, \kappa \in (0,1)$, *for the BS-BP, the BS-DS, and the BS-LASSO, respectively. Define* $\boldsymbol{h} = \hat{\boldsymbol{x}} - \boldsymbol{x}$ *as the error vector for any of the three block-*ℓ_1 *recovery algorithms* (5.2), (5.3), *and* (5.4)*. Then we have*

$$\|\boldsymbol{h}_{[S]}\|_{\mathrm{b}1} \geq \|\boldsymbol{h}\|_{\mathrm{b}1}/c, \tag{6.1}$$

where $S = \mathrm{bsupp}(\boldsymbol{x})$, $c = 2$ *for the BS-BP and the BS-DS, and* $c = 2/(1-\kappa)$ *for the BS-LASSO.*

Proof. Suppose $S = \mathrm{bsupp}(\boldsymbol{x})$ and $|S| = \|\boldsymbol{x}\|_{\mathrm{b}0} = k$. Define the error vector $\boldsymbol{h} = \hat{\boldsymbol{x}} - \boldsymbol{x}$.

We first prove the proposition for the BS-BP and the BS-DS. The fact that $\|\hat{\boldsymbol{x}}\|_{\mathrm{b}1} = \|\boldsymbol{x} + \boldsymbol{h}\|_{\mathrm{b}1}$ is the minimum among all $\boldsymbol{z}$s satisfying the constraints in (5.2) and (5.3), together with the fact that the true signal $\boldsymbol{x}$ satisfies the constraints as required by the conditions imposed on the noise in Proposition 6.1.1, implies that $\|\boldsymbol{h}_{[S^c]}\|_{\mathrm{b}1}$ cannot be very large. To see this, note that

$$\begin{aligned}\|\boldsymbol{x}\|_{\mathrm{b}1} &\geq \|\boldsymbol{x}+\boldsymbol{h}\|_{\mathrm{b}1} \\ &= \sum_{i\in S}\|\boldsymbol{x}_{[i]}+\boldsymbol{h}_{[i]}\|_2+\sum_{i\in S^c}\|\boldsymbol{x}_{[i]}+\boldsymbol{h}_{[i]}\|_2 \\ &\geq \sum_{i\in S}\|\boldsymbol{x}_{[i]}\|_2-\sum_{i\in S}\|\boldsymbol{h}_{[i]}\|_2+\sum_{i\in S^c}\|\boldsymbol{h}_{[i]}\|_2 \\ &= \|\boldsymbol{x}_{[S]}\|_{\mathrm{b}1}-\|\boldsymbol{h}_{[S]}\|_{\mathrm{b}1}+\|\boldsymbol{h}_{[S^c]}\|_{\mathrm{b}1} \\ &= \|\boldsymbol{x}\|_{\mathrm{b}1}-\|\boldsymbol{h}_{[S]}\|_{\mathrm{b}1}+\|\boldsymbol{h}_{[S^c]}\|_{\mathrm{b}1}. \end{aligned} \tag{6.2}$$

Therefore, we obtain $\|\boldsymbol{h}_{[S]}\|_{\mathrm{b}1} \geq \|\boldsymbol{h}_{[S^c]}\|_{\mathrm{b}1}$, which leads to

$$2\|\boldsymbol{h}_{[S]}\|_{\mathrm{b}1} \geq \|\boldsymbol{h}_{[S]}\|_{\mathrm{b}1}+\|\boldsymbol{h}_{[S^c]}\|_{\mathrm{b}1}=\|\boldsymbol{h}\|_{\mathrm{b}1}. \tag{6.3}$$

We now turn to the BS-LASSO (5.4). Since the noise $\boldsymbol{w}$ satisfies $\|A^T\boldsymbol{w}\|_{\mathrm{b}\infty} \leq \kappa\mu$ for some $\kappa \in (0,1)$, and $\hat{\boldsymbol{x}}$ is the minimizer of (5.4), we have

$$\frac{1}{2}\|A\hat{\boldsymbol{x}}-\boldsymbol{y}\|_2^2+\mu\|\hat{\boldsymbol{x}}\|_{\mathrm{b}1} \leq \frac{1}{2}\|A\boldsymbol{x}-\boldsymbol{y}\|_2^2+\mu\|\boldsymbol{x}\|_{\mathrm{b}1}.$$

Consequently, substituting $\boldsymbol{y}=A\boldsymbol{x}+\boldsymbol{w}$ yields

$$\begin{aligned}\mu\|\hat{\boldsymbol{x}}\|_{\mathrm{b}1} &\leq \frac{1}{2}\|A\boldsymbol{x}-\boldsymbol{y}\|_2^2-\frac{1}{2}\|A\hat{\boldsymbol{x}}-\boldsymbol{y}\|_2^2+\mu\|\boldsymbol{x}\|_{\mathrm{b}1} \\ &= \frac{1}{2}\|\boldsymbol{w}\|_2^2-\frac{1}{2}\|A(\hat{\boldsymbol{x}}-\boldsymbol{x})-\boldsymbol{w}\|_2^2+\mu\|\boldsymbol{x}\|_{\mathrm{b}1} \\ &= \frac{1}{2}\|\boldsymbol{w}\|_2^2-\frac{1}{2}\|A(\hat{\boldsymbol{x}}-\boldsymbol{x})\|_2^2 \\ &\qquad +\langle A(\hat{\boldsymbol{x}}-\boldsymbol{x}),\boldsymbol{w}\rangle-\frac{1}{2}\|\boldsymbol{w}\|_2^2+\mu\|\boldsymbol{x}\|_{\mathrm{b}1} \\ &\leq \langle A(\hat{\boldsymbol{x}}-\boldsymbol{x}),\boldsymbol{w}\rangle+\mu\|\boldsymbol{x}\|_{\mathrm{b}1} \\ &= \left\langle \hat{\boldsymbol{x}}-\boldsymbol{x},A^T\boldsymbol{w}\right\rangle+\mu\|\boldsymbol{x}\|_{\mathrm{b}1}. \end{aligned}$$

Using the Cauchy-Swcharz type inequality, we get

$$\begin{aligned}\mu\|\hat{\boldsymbol{x}}\|_{\mathrm{b}1} &\leq \|\hat{\boldsymbol{x}}-\boldsymbol{x}\|_{\mathrm{b}1}\|A^T\boldsymbol{w}\|_{\mathrm{b}\infty}+\mu\|\boldsymbol{x}\|_{\mathrm{b}1} \\ &= \kappa\mu\|\boldsymbol{h}\|_{\mathrm{b}1}+\mu\|\boldsymbol{x}\|_{\mathrm{b}1}, \end{aligned}$$

which leads to

$$\|\hat{\boldsymbol{x}}\|_{\mathrm{b}1} \;\; \leq \;\; \kappa\|\boldsymbol{h}\|_{\mathrm{b}1} + \|\boldsymbol{x}\|_{\mathrm{b}1}.$$

Therefore, similar to the argument in (6.2), we have

$$\begin{aligned}
& \|\boldsymbol{x}\|_{\mathrm{b}1} \\
\geq \;\; & \|\hat{\boldsymbol{x}}\|_{\mathrm{b}1} - \kappa\|\boldsymbol{h}\|_{\mathrm{b}1} \\
= \;\; & \|\boldsymbol{x} + \boldsymbol{h}_{[S^c]} + \boldsymbol{h}_{[S]}\|_{\mathrm{b}1} - \kappa\left(\|\boldsymbol{h}_{[S^c]} + \boldsymbol{h}_{[S]}\|_{\mathrm{b}1}\right) \\
\geq \;\; & \|\boldsymbol{x} + \boldsymbol{h}_{[S^c]}\|_{\mathrm{b}1} - \|\boldsymbol{h}_{[S]}\|_{\mathrm{b}1} - \kappa\left(\|\boldsymbol{h}_{[S^c]}\|_{\mathrm{b}1} + \|\boldsymbol{h}_{[S]}\|_{\mathrm{b}1}\right) \\
= \;\; & \|\boldsymbol{x}\|_{\mathrm{b}1} + (1-\kappa)\|\boldsymbol{h}_{[S^c]}\|_{\mathrm{b}1} - (1+\kappa)\|\boldsymbol{h}_{[S]}\|_{\mathrm{b}1},
\end{aligned}$$

where $S = \mathrm{bsupp}(\boldsymbol{x})$. Consequently, we have

$$\|\boldsymbol{h}_{[S]}\|_{\mathrm{b}1} \;\; \geq \;\; \frac{1-\kappa}{1+\kappa}\|\boldsymbol{h}_{[S^c]}\|_{\mathrm{b}1}.$$

Therefore, similar to (6.3), we obtain

$$\begin{aligned}
\frac{2}{1-\kappa}\|\boldsymbol{h}_{[S]}\|_{\mathrm{b}1} \;\; &= \;\; \frac{1+\kappa}{1-\kappa}\|\boldsymbol{h}_{[S]}\|_{\mathrm{b}1} + \frac{1-\kappa}{1-\kappa}\|\boldsymbol{h}_{[S]}\|_{\mathrm{b}1} \\
&\geq \;\; \frac{1+\kappa}{1-\kappa}\frac{1-\kappa}{1+\kappa}\|\boldsymbol{h}_{[S^c]}\|_{\mathrm{b}1} + \frac{1-\kappa}{1-\kappa}\|\boldsymbol{h}_{[S]}\|_{\mathrm{b}1} \\
&= \;\; \|\boldsymbol{h}\|_{\mathrm{b}1}. \qquad (6.4)
\end{aligned}$$

□

An immediate corollary of Proposition 6.1.1 is to bound the block-ℓ_1 and ℓ_2 norms of the error vectors using the block-ℓ_∞ norm.

Corollary 6.1.1. *Under the assumptions of Proposition 6.1.1, we have*

$$\begin{aligned}
\|\boldsymbol{h}\|_{\mathrm{b}1} \;\; &\leq \;\; ck\|\boldsymbol{h}\|_{\mathrm{b}\infty}, \qquad (6.5) \\
\|\boldsymbol{h}\|_2 \;\; &\leq \;\; \sqrt{ck}\|\boldsymbol{h}\|_{\mathrm{b}\infty}. \qquad (6.6)
\end{aligned}$$

Furthermore, if $S = \mathrm{bsupp}(\boldsymbol{x})$ *and* $\beta = \min_{i\in S}\|\boldsymbol{x}_{[i]}\|_2$*, then* $\|\boldsymbol{h}\|_{\mathrm{b}\infty} < \beta/2$ *implies*

$$\{i : \|\hat{\boldsymbol{x}}_{[i]}\|_2 > \beta/2\} \;\; = \;\; \mathrm{bsupp}(\boldsymbol{x}), \qquad (6.7)$$

i.e., *a thresholding operator recovers the signal block-support.*

Proof. Suppose $S = \mathrm{bsupp}(\boldsymbol{x})$. According to Proposition 6.1.1, we have

$$\|\boldsymbol{h}\|_{\mathrm{b}1} \leq c\|\boldsymbol{h}_{[S]}\|_{\mathrm{b}1} \leq ck\|\boldsymbol{h}\|_{\mathrm{b}\infty}. \tag{6.8}$$

To prove (6.6), we note

$$\begin{aligned}
\frac{\|\boldsymbol{h}\|_2^2}{\|\boldsymbol{h}\|_{\mathrm{b}\infty}^2} &= \sum_{i=1}^{p} \left(\frac{\|\boldsymbol{h}_{[i]}\|_2}{\|\boldsymbol{h}\|_{\mathrm{b}\infty}}\right)^2 \\
&\leq \sum_{i=1}^{p} \left(\frac{\|\boldsymbol{h}_{[i]}\|_2}{\|\boldsymbol{h}\|_{\mathrm{b}\infty}}\right) \\
&= \frac{\|\boldsymbol{h}\|_{\mathrm{b}1}}{\|\boldsymbol{h}\|_{\mathrm{b}\infty}} \\
&\leq ck.
\end{aligned} \tag{6.9}$$

For the first inequality, we have used $\frac{\|\boldsymbol{h}_{[i]}\|_2}{\|\boldsymbol{h}\|_{\mathrm{b}\infty}} \leq 1$ and $a^2 \leq a$ for $a \in [0, 1]$.

For the last assertion, note that if $\|\boldsymbol{h}\|_{\mathrm{b}\infty} \leq \beta/2$, then we have for $i \in S$,

$$\|\hat{\boldsymbol{x}}_{[i]}\|_2 = \|\boldsymbol{x}_{[i]} + \boldsymbol{h}_{[i]}\|_2 \geq \|\boldsymbol{x}_{[i]}\|_2 - \|\boldsymbol{h}_{[i]}\|_2 > \beta - \beta/2 = \beta/2; \tag{6.10}$$

and for $i \notin S$,

$$\|\hat{\boldsymbol{x}}_{[i]}\|_2 = \|\boldsymbol{x}_{[i]} + \boldsymbol{h}_{[i]}\|_2 = \|\boldsymbol{h}_{[i]}\|_2 < \beta/2. \tag{6.11}$$

□

For ease of presentation, we introduce the following notation:

Definition 6.1.1. *For any $s \in [1, p]$ and matrix $A \in \mathbb{R}^{m \times np}$, define*

$$\omega_\diamond(Q, s) = \min_{\boldsymbol{z}:\|\boldsymbol{z}\|_{\mathrm{b}1}/\|\boldsymbol{z}\|_{\mathrm{b}\infty} \leq s} \frac{\|Q\boldsymbol{z}\|_\diamond}{\|\boldsymbol{z}\|_{\mathrm{b}\infty}}, \tag{6.12}$$

where Q is either A or $A^T A$.

Now we present the error bounds on the block-ℓ_∞ norm of the error vectors for the BS-BP, the BS-DS, and the BS-LASSO.

Theorem 6.1.1. *Under the assumption of Proposition 6.1.1, we have*

$$\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_{\mathrm{b}\infty} \leq \frac{2\varepsilon}{\omega_2(A, 2k)} \tag{6.13}$$

for the BS-BP,

$$\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_{\mathrm{b}\infty} \leq \frac{2\mu}{\omega_{\mathrm{b}\infty}(A^T A, 2k)} \tag{6.14}$$

for the BS-DS, and

$$\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_{\mathrm{b}\infty} \leq \frac{(1+\kappa)\mu}{\omega_{\mathrm{b}\infty}(A^T A, 2k/(1-\kappa))} \tag{6.15}$$

for the BS-LASSO.

Proof. Observe that for the BS-BP

$$\begin{aligned} \|A(\hat{\boldsymbol{x}} - \boldsymbol{x})\|_2 &\leq \|\boldsymbol{y} - A\hat{\boldsymbol{x}}\|_2 + \|\boldsymbol{y} - A\boldsymbol{x}\|_2 \\ &\leq \varepsilon + \|A\boldsymbol{w}\|_2 \\ &\leq 2\varepsilon, \end{aligned} \tag{6.16}$$

and similarly,

$$\|A^T A(\hat{\boldsymbol{x}} - \boldsymbol{x})\|_{\mathrm{b}\infty} \leq 2\mu \tag{6.17}$$

for the BS-DS, and

$$\|A^T A(\hat{\boldsymbol{x}} - \boldsymbol{x})\|_{\mathrm{b}\infty} \leq (1+\kappa)\mu \tag{6.18}$$

for the BS-LASSO. Here for the BS-LASSO, we have used

$$\|A^T(A\hat{\boldsymbol{x}} - \boldsymbol{y})\|_{\mathrm{b}\infty} \leq \mu, \tag{6.19}$$

a consequence of the optimality condition

$$A^T(A\hat{\boldsymbol{x}} - \boldsymbol{y}) \in \mu \partial \|\hat{\boldsymbol{x}}\|_{\mathrm{b}1} \tag{6.20}$$

and the fact that the ith block of any subgradient in $\partial\|\hat{\boldsymbol{x}}\|_{\mathrm{b}1}$ is $\hat{\boldsymbol{x}}_{[i]}/\|\hat{\boldsymbol{x}}_{[i]}\|_2$ if $\hat{\boldsymbol{x}}_{[i]} \neq 0$ and is $\boldsymbol{g}$ otherwise for some $\|\boldsymbol{g}\|_2 \leq 1$.

The conclusions of Theorem 6.1.1 follow from equation (6.5) and Definition 6.1.1. □

A consequence of Theorem 6.1.1 and Corollary 6.1.1 is the error bound on the ℓ_2 norm:

Corollary 6.1.2. *Under the assumption of Proposition 6.1.1, the ℓ_2 norms of the recovery errors are bounded as*

$$\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2 \leq \frac{2\sqrt{2k}\varepsilon}{\omega_2(A, 2k)} \tag{6.21}$$

for the BS-BP,

$$\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2 \leq \frac{2\sqrt{2k}\mu}{\omega_{\mathrm{b}\infty}(A^TA, 2k)} \tag{6.22}$$

for the BS-DS, and

$$\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2 \leq \sqrt{\frac{2k}{1-\kappa}} \frac{(1+\kappa)\mu}{\omega_{\mathrm{b}\infty}(A^TA, 2k/(1-\kappa))} \tag{6.23}$$

for the BS-LASSO.

Although the results presented so far are in strict parallel with those for sparsity recovery, it is difficult to design algorithms to compute exact $\omega_\diamond(A, s)$ and $\omega_{\mathrm{b}\infty}(A^TA, s)$. We will design algorithm to lower bound $\omega_\diamond(A, s)$ and $\omega_{\mathrm{b}\infty}(A^TA, s)$. When these lower bounds are in place of $\omega_\diamond(A, s)$ and $\omega_{\mathrm{b}\infty}(A^TA, s)$ in Theorem 6.1.1, the resulting expressions are genuine upper bounds on the recovery errors for block-sparsity recovery. Therefore, the algorithms provide a way to numerically assess the performance of the BS-BP, the BS-DS, and the BS-LASSO. According to Corollary 6.1.1, the correct recovery of signal block-support is also guaranteed by reducing the block-ℓ_∞ norm to some threshold. In Section 6.4, we also demonstrate that the bounds in Theorem 6.1.1 are non-trivial for a large class of random sensing matrices, as long as m is relatively large. Our preliminary numerical simulations in Section 6.5 show that in many cases the error bounds on the ℓ_2 norms based on Corollary 6.1.2 are tighter than the block RIP based bounds.

6.2 Verification Algorithm

In this section, we consider the computational issues of $\omega_\diamond(\cdot)$.

6.2.1 Semidefinite Relaxation for Verification

A prerequisite for the bounds in Theorem 6.1.1 to be valid is the positiveness of the involved $\omega_\diamond(\cdot)$. We call the validation of $\omega_\diamond(\cdot) > 0$ the verification problem. Note that from Theorem 6.1.1, $\omega_\diamond(\cdot) > 0$ implies the exact recovery of the true signal $\boldsymbol{x}$ in the noise-free case. Therefore, verifying $\omega_\diamond(\cdot) > 0$ is equivalent to verifying a sufficient condition for exact block-ℓ_1 recovery.

Verifying $\omega_\diamond(Q, s) > 0$ is equivalent with making sure $\|\boldsymbol{z}\|_{\mathrm{b}1}/\|\boldsymbol{z}\|_{\mathrm{b}\infty} \leq s$ for all $\boldsymbol{z}$ such that $Q\boldsymbol{z} = 0$. Therefore, we compute

$$s^* \quad = \quad \min_{\boldsymbol{z}} \frac{\|\boldsymbol{z}\|_{\mathrm{b}1}}{\|\boldsymbol{z}\|_{\mathrm{b}\infty}} \text{ s.t. } \quad Q\boldsymbol{z} = 0. \tag{6.24}$$

Then, when $s < s^*$, we have $\omega_\diamond(Q, s) > 0$. The following theorem presents an optimization procedure that computes a lower bound on s^*.

Proposition 6.2.1. *The reciprocal of the optimal value of the following optimization, denoted by s_*,*

$$\max_{i} \min_{P_{[i]}} \max_{j} \|\delta_{ij}\mathrm{I}_n - P_{[i]}^T Q_{[j]}\|_2 \tag{6.25}$$

is a lower bound on s^**. Here* P *is a matrix variable of the same size as* Q*,* $\delta_{ij} = 1$ *for* $i = j$ *and* 0 *otherwise, and* $P = \left[P_{[1]}, \ldots, P_{[p]}\right]$*,* $Q = \left[Q_{[1]}, \ldots, Q_{[p]}\right]$ *with* $P_{[i]}$ *and* $Q_{[j]}$ *having* n *columns each.*

Proof. We rewrite the optimization (6.24) as

$$\frac{1}{s^*} = \max_{\boldsymbol{z}} \|\boldsymbol{z}\|_{\mathrm{b}\infty} \text{ s.t. } \quad Q\boldsymbol{z} = 0, \|\boldsymbol{z}\|_{\mathrm{b}1} \leq 1. \tag{6.26}$$

Note that in (6.26), we are maximizing a convex function over a convex set, which is in general very difficult. We will use a relaxation technique to compute an upper bound on the optimal value of (6.26). Define a matrix variable P of the same size as Q. Since the dual norm of $\|\cdot\|_{\mathrm{b}\infty}$ is $\|\cdot\|_{\mathrm{b}1}$, we have

$$\begin{aligned}
& \max_{\boldsymbol{z}} \{\|\boldsymbol{z}\|_{\mathrm{b}\infty} : \|\boldsymbol{z}\|_{\mathrm{b}1} \leq 1, Q\boldsymbol{z} = 0\} \\
= \;& \max_{\boldsymbol{u},\boldsymbol{z}} \left\{\boldsymbol{u}^T\boldsymbol{z} : \|\boldsymbol{z}\|_{\mathrm{b}1} \leq 1, \|\boldsymbol{u}\|_{\mathrm{b}1} \leq 1, Q\boldsymbol{z} = 0\right\} \\
= \;& \max_{\boldsymbol{u},\boldsymbol{z}} \left\{\boldsymbol{u}^T(\boldsymbol{z} - P^TQ\boldsymbol{z}) : \|\boldsymbol{z}\|_{\mathrm{b}1} \leq 1, \|\boldsymbol{u}\|_{\mathrm{b}1} \leq 1, Q\boldsymbol{z} = 0\right\} \\
\leq \;& \max_{\boldsymbol{u},\boldsymbol{z}} \left\{\boldsymbol{u}^T(\mathrm{I}_{np} - P^TQ)\boldsymbol{z} : \|\boldsymbol{z}\|_{\mathrm{b}1} \leq 1, \|\boldsymbol{u}\|_{\mathrm{b}1} \leq 1\right\}.
\end{aligned} \tag{6.27}$$

In the last expression, we have dropped the constraint $Q\boldsymbol{z} = 0$. Note that the unit ball $\{\boldsymbol{z} : \|\boldsymbol{z}\|_{\mathrm{b}1} \leq 1\} \subset \mathbb{R}^{np}$ is the convex hall of $\{\mathbf{e}_i^p \otimes \boldsymbol{v} : 1 \leq i \leq p, \boldsymbol{v} \in \mathbb{R}^n, \|\boldsymbol{v}\|_2 \leq 1\}$ and $\boldsymbol{u}^T(\mathrm{I}_{np} - P^TQ)\boldsymbol{z}$ is convex (actually, linear) in $\boldsymbol{z}$. As a consequence, we have

$$\begin{aligned}
& \max_{\boldsymbol{u},\boldsymbol{z}} \left\{\boldsymbol{u}^T(\mathrm{I}_{np} - P^TQ)\boldsymbol{z} : \|\boldsymbol{z}\|_{\mathrm{b}1} \leq 1, \|\boldsymbol{u}\|_{\mathrm{b}1} \leq 1\right\} \\
= \;& \max_{j,\boldsymbol{v},\boldsymbol{u}} \left\{\boldsymbol{u}^T(\mathrm{I}_{np} - P^TQ)(\mathbf{e}_i^p \otimes \boldsymbol{v}) : \|\boldsymbol{u}\|_{\mathrm{b}1} \leq 1, \|\boldsymbol{v}\|_2 \leq 1\right\} \\
= \;& \max_{j} \max_{\boldsymbol{v},\boldsymbol{u}} \left\{\boldsymbol{u}^T(\mathrm{I}_{np} - P^TQ)_{[j]}\boldsymbol{v} : \|\boldsymbol{u}\|_{\mathrm{b}1} \leq 1, \|\boldsymbol{v}\|_2 \leq 1\right\} \\
= \;& \max_{j} \max_{\boldsymbol{u}} \left\{\|(\mathrm{I}_{np} - P^TQ)_{[j]}^T\boldsymbol{u}\|_2 : \|\boldsymbol{u}\|_{\mathrm{b}1} \leq 1\right\},
\end{aligned} \tag{6.28}$$

where $(\mathrm{I}_{np} - P^TQ)_{[j]}$ denotes the jth column blocks of $\mathrm{I}_{np} - P^TQ$, namely, the submatrix of $\mathrm{I}_{np} - P^TQ$ formed by the $((j-1)n + 1)$th to jnth columns.

Applying the same argument to the unit ball $\{\boldsymbol{u} : \|\boldsymbol{u}\|_{\mathrm{b}1} \leq 1\}$ and the convex function $\|(\mathrm{I}_{np} - P^TQ)_{[j]}^T \boldsymbol{u}\|_2$, we obtain

$$\begin{aligned} & \max_{\boldsymbol{u},\boldsymbol{z}} \left\{ \boldsymbol{u}^T(\mathrm{I}_{np} - P^TQ)\boldsymbol{z} : \|\boldsymbol{z}\|_{\mathrm{b}1} \leq 1, \|\boldsymbol{u}\|_{\mathrm{b}1} \leq 1 \right\} \\ = \ & \max_{i,j} \|(\mathrm{I}_{np} - P^TQ)_{[i],[j]}\|_2. \end{aligned} \tag{6.29}$$

Here $(\mathrm{I}_{np} - P^TQ)_{i,j}$ is the submatrix of $\mathrm{I}_{np} - P^TQ$ formed by the $((i-1)n+1)$th to inth rows and the $((j-1)n+1)$th to jnth columns, and $\|\cdot\|_2$ is the spectral norm (the largest singular value).

Since P is arbitrary, the tightest upper bound is obtained by minimizing $\max_{i,j} \|(\mathrm{I}_{np} - P^TQ)_{[i],[j]}\|_2$ with respect to P:

$$\begin{aligned} 1/s^* &= \max_{\boldsymbol{z}} \{\|\boldsymbol{z}\|_{\mathrm{b}\infty} : \|\boldsymbol{z}\|_{\mathrm{b}1} \leq 1, Q\boldsymbol{z} = 0\} \\ &\leq \min_P \max_{i,j} \|(\mathrm{I}_{np} - P^TQ)_{[i],[j]}\|_2 \\ &= 1/s_*. \end{aligned} \tag{6.30}$$

Partition P and Q as $P = \left[P_{[1]}, \ldots, P_{[p]}\right]$ and $Q = \left[Q_{[1]}, \ldots, Q_{[p]}\right]$ with $P_{[i]}$ and $Q_{[j]}$ having n columns each. We explicitly write

$$(\mathrm{I}_{np} - P^TQ)_{[i],[j]} = \delta_{ij}\mathrm{I}_n - P_{[i]}^T Q_{[j]}, \tag{6.31}$$

where $\delta_{ij} = 1$ for $i = j$ and 0 otherwise. As a consequence, we obtain

$$\begin{aligned} & \min_P \max_{i,j} \|(\mathrm{I}_{np} - P^TQ)_{[i],[j]}\|_2 \\ = \ & \min_{P_{[1]},\ldots,P_{[p]}} \max_i \max_j \|\delta_{ij}\mathrm{I}_n - P_{[i]}^T Q_{[j]}\|_2 \\ = \ & \max_i \min_{P_{[i]}} \max_j \|\delta_{ij}\mathrm{I}_n - P_{[i]}^T Q_{[j]}\|_2. \end{aligned} \tag{6.32}$$

We have moved the $\max_i$ to the outmost because for each i, $\max_j \|\delta_{ij}\mathrm{I}_n - P_{[i]}^T Q_{[j]}\|_2$ is a function of only $P_{[i]}$ and does not depends on other variables $P_{[l]}, l \neq i$. $\square$

Because $s_* < s^*$, the condition $s < s_*$ is sufficient condition for $\omega_\diamond > 0$ and for the uniqueness and exactness of block-sparse recovery in the noise free case. To get s_*, for each i, we need to solve

$$\min_{P_{[i]}} \max_{j} \|\delta_{ij}\mathrm{I}_n - P_{[i]}^T Q_{[j]}\|_2. \tag{6.33}$$

An equivalent semidefinite program is obtained as follows:

$$\min_{P_{[i]},t} t \text{ s.t. } \|\delta_{ij}\mathrm{I}_n - P_{[i]}^T Q_{[j]}\|_2 \leq t, j = 1, \ldots, p. \tag{6.34}$$

$$\Leftrightarrow \min_{P_{[i]},t} t \text{ s.t. } \begin{bmatrix} t\mathrm{I}_n & \delta_{ij}\mathrm{I}_n - P_{[i]}^T Q_{[j]} \\ \delta_{ij} I_n - Q_{[j]}^T P_{[i]} & t\mathrm{I}_n \end{bmatrix} \succeq 0, j = 1, \ldots, p. \tag{6.35}$$

Small instances of (6.34) and (6.35) can be solved using CVX [91]. However, it is beneficial to use first-order techniques to solve (6.33) directly. We observe that $\max_j \|\delta_{ij}\mathrm{I}_n - P_{[i]}^T Q_{[j]}\|_2$ can be expressed as the largest eigenvalue of a block-diagonal matrix ($\mathcal{Q}(P_{[i]})$ defined in (6.37)). The smoothing technique for semidefinite optimization developed in [92] is then used to minimize the largest eigenvalue of $\mathcal{Q}(P_{[i]})$ with respect to $P_{[i]}$. To get an ϵ accuracy, the overall time complexity for computing (6.25) is $O\left(n^3 p^2 \sqrt{\log np}/\epsilon\right)$. We present more details in the next subsection.

6.2.2 Smoothing Technique for Solving (6.33)

We first consider the problem (6.33). Since for any matrix X, the non-zero eigenvalues of

$$\begin{bmatrix} \mathbf{0} & X \\ X^T & \mathbf{0} \end{bmatrix} \tag{6.36}$$

are $\{\sigma_l(X)\} \bigcup \{-\sigma_l(X)\}$, where $\sigma_l(X)$ is the lth non-zero singular value of X, we could control the largest singular value of X by controlling the largest eigenvalue of (6.36). In addition, the eigenvalues of a block diagonal matrix is the union of the eigenvalues of its blocks. As a consequence, $\min_{P_{[i]}} \max_j \|\delta_{ij}\mathrm{I}_n - P_{[i]}^T Q_{[j]}\|_2$ is equivalent to minimizing the largest eigenvalue of

$$\begin{aligned}\mathcal{Q}(P_{[i]}) &\overset{\text{def}}{=} \begin{bmatrix} \mathbf{0} & \delta_{i1}\mathrm{I}_n - P_{[i]}^T Q_{[1]} & & & \\ \delta_{i1}\mathrm{I}_n - Q_{[1]}^T P_{[i]} & \mathbf{0} & & & \\ & & \ddots & & \\ & & & \mathbf{0} & \delta_{ip}\mathrm{I}_n - P_{[i]}^T Q_{[p]} \\ & & & \delta_{ip}\mathrm{I}_n - Q_{[p]}^T P_{[i]} & \mathbf{0} \end{bmatrix} \\ &\overset{\text{def}}{=} \begin{bmatrix} \mathcal{Q}_1(P_{[i]}) & & \\ & \ddots & \\ & & \mathcal{Q}_p(P_{[i]}) \end{bmatrix} \end{aligned} \tag{6.37}$$

with respect to $P_{[i]}$. We employ a first order smoothing technique developed in [92] to solve the minimization of largest eigenvalue problem

$$\min_P \lambda_{\max}(\mathcal{Q}(P)), \tag{6.38}$$

where for notation simplicity we have omitted the subscript i.

For any $\mu > 0$, define the following smooth approximation of maximal eigenvalue function

$$\begin{aligned}\phi_\mu(P) &\overset{\text{def}}{=} \mu \log \left(\operatorname{trace} \exp \left(\frac{\mathcal{Q}(P)}{\mu} \right) \right) \\ &\overset{\text{def}}{=} \mu \log F_\mu(P), \end{aligned} \tag{6.39}$$

which satisfies

$$\lambda_{\max}(\mathcal{Q}(P)) \le \phi_\mu(P) \le \lambda_{\max}(\mathcal{Q}(P)) + \mu \log(2np). \tag{6.40}$$

Thus, if $\mu = \epsilon/(2\log(2np))$ with $\epsilon > 0$ the target precision, then $\phi_\mu(P)$ is a $\epsilon/2$ approximation of $\lambda_{\max}(\mathcal{Q}(P))$. In addition, since whenever $\|P\|_{\mathrm{F}} \leq 1$, we have

$$\begin{aligned} & \max_j \left\| \begin{bmatrix} \mathbf{0} & -P^T Q_{[j]} \\ -Q_{[j]}^T P & \mathbf{0} \end{bmatrix} \right\|_2^2 \\ = \; & \max_j \|P^T Q_{[j]}\|_2^2 \\ \leq \; & \max_j \|P\|_2^2 \|Q_{[j]}\|_2^2 \\ \leq \; & \max_j \|Q_{[j]}\|_2^2. \end{aligned} \tag{6.41}$$

according to [92, Section 4], the gradient of $\phi_\mu(P)$ is Lipschitz continuous with respect to the Frobenius norm and has Lipschitz constant

$$L = \frac{2\log(2np)\max_j \|Q_{[j]}\|_2^2}{\epsilon}. \tag{6.42}$$

Note that for (6.33) we could always take Q as the orthogonal matrix obtained through the QR decomposition of A^T. As a consequence, $\max_j \|Q_{[j]}\|_2 \leq 1$ and we take

$$L = \frac{2\log(2np)}{\epsilon}. \tag{6.43}$$

We define the gradient mapping $T(P) = P - L^{-1}\nabla\phi_\mu(P)$ as the optimal solution to the following minimization problem:

$$\min_R \left\{ \langle \nabla\phi_\mu(P), R - P \rangle + \frac{1}{2} L \|R - P\|_{\mathrm{F}}^2 \right\}, \tag{6.44}$$

and define the prox-function $d(P) = \|P\|_{\mathrm{F}}^2/2$. The algorithm in [92] for solving (6.33) goes as

For $k \geq 0$ **do**

1. Compute $\phi_\mu(P^{(k)})$ and $\nabla\phi_\mu(P^{(k)})$
2. Find $U^{(k)} = T(P^{(k)}) = P^{(k)} - L^{-1}\nabla\phi_\mu(P^{(k)})$.

3. Find

$$\begin{aligned} V^{(k)} &= \operatorname{argmin}_V \left\{ Ld(V) + \sum_{i=1}^{k} \frac{i+1}{2} \left[\phi_\mu(P^{(i)}) + \left\langle \nabla\phi_\mu(P^{(i)}), V - P^{(i)} \right\rangle \right] \right\} \\ &= -\frac{1}{L} \sum_{i=0}^{k} \frac{i+1}{2} \nabla\phi_\mu(P^{(i)}) \end{aligned}$$

4. Set $P^{(k+1)} = \frac{2}{k+3} V^{(k)} + \frac{k+1}{k+3} U^{(k)}$.

The most expensive step of the algorithm is computing the matrix exponential. However, we could alleviate this by exploiting the block-diagonal structure of $\mathcal{Q}(P)$. More precisely, we have

$$\begin{aligned} F_\mu(P) &= \operatorname{trace} \exp\left(\frac{\mathcal{Q}(P)}{\mu} \right) \\ &= \sum_{j=1}^{p} \operatorname{trace} \exp\left(\frac{\mathcal{Q}_{[j]}(P)}{\mu} \right), \end{aligned} \tag{6.45}$$

and

$$\begin{aligned} \phi_\mu &= \mu \log F_\mu(P) \\ &= \mu \log \left[\sum_{j=1}^{p} \operatorname{trace} \exp\left(\frac{\mathcal{Q}_{[j]}(P)}{\mu} \right) \right]. \end{aligned} \tag{6.46}$$

The exponential of $\mathcal{Q}(P)/\mu$ is also a block-diagonal matrix, whose blocks are obtained by computing the exponential of the blocks in the diagonal of $\mathcal{Q}(P)/\mu$. The total complexity for computing $F_\mu(P)$ is $O(n^3 p)$.

It remains to derive a formula for $\nabla\phi_\mu(P)$, the key of which is to compute the derivative of the trace exponential function. We need the following two lemmas.

Lemma 6.2.1. *Define* $p(t) = \operatorname{trace}(A + tB)^k$. *Then we have* $p'(t) = k \operatorname{trace}((A + tB)^{k-1} B)$.

Proof. Since $p(t) = \operatorname{trace}\left((A + t_0B) + (t - t_0)B\right)^k$, we only need to prove the result for $t = 0$. We note

$$p(t) \;=\; \sum_{j=0}^{k} t^j \operatorname{trace}(S_{k,j}(A, B)), \tag{6.47}$$

where the Hurwitz product $S_{k,j}(A, B)$ is the sum of all words of length k in A and B and j B's appear. In addition, we have $S_{k,0}(A, B) = A^k$, and $\operatorname{trace}(S_{k,1}(A, B)) = k\operatorname{trace}(A^{k-1}B)$ due to $\operatorname{trace}(CD) = \operatorname{trace}(DC)$ for any C and D of compatible dimensions. As a consequence, we obtain

$$\begin{aligned}
p'(0) &= \lim_{t\to 0} \frac{p(t) - p(0)}{t} \\
&= \lim_{t\to 0} \frac{\left(A^k + tk\operatorname{trace}(A^{k-1}B) + t^2 \sum_{j=2}^{k} t^{j-2} S_{k,j}(A, B)\right) - A^k}{t} \\
&= \lim_{t\to 0} k\operatorname{trace}(A^{k-1}B) + t \sum_{j=2}^{k} t^{j-2} S_{k,j}(A, B) \\
&= k\operatorname{trace}(A^{k-1}B). \qquad (6.48)
\end{aligned}$$

□

Lemma 6.2.2. *Define* $q(t) = \operatorname{trace}\exp(A+tB)$. *Then we have* $q'(t) = \operatorname{trace}\left(\exp(A + tB)B\right)$.

Proof. The calculation proceeds as follows:

$$\begin{aligned}
q'(t) &= \left(\sum_{k=0}^{\infty} \frac{\operatorname{trace}(A + tB)^k}{k!}\right)' \\
&= \left(\sum_{k=0}^{\infty} \frac{\left[\operatorname{trace}(A + tB)^k\right]'}{k!}\right) \\
&= \left(\sum_{k=0}^{\infty} \frac{\operatorname{trace}\left((A + tB)^k B\right)}{k!}\right) \\
&= \operatorname{trace}\left(\sum_{k=0}^{\infty} \frac{(A + tB)^k}{k!} B\right) \\
&= \operatorname{trace}\left(\exp(A + tB)B\right). \qquad (6.49)
\end{aligned}$$

□

We now compute $\nabla\phi_\mu(P) = \frac{\mu}{F_\mu(P)}\nabla F_\mu(P)$. Note that

$$\begin{aligned}\nabla F_\mu(P) &= \nabla \mathrm{trace}\exp\left(\frac{\mathcal{Q}(P)}{\mu}\right)\\ &= \sum_{j=1}^{p}\nabla \mathrm{trace}\exp\left(\frac{\mathcal{Q}_{[j]}(P)}{\mu}\right). \end{aligned} \tag{6.50}$$

The partial derivative is computed as follows

$$\begin{aligned}&\nabla_{P_{\alpha\beta}}\mathrm{trace}\exp\left(\frac{\mathcal{Q}_{[j]}(P)}{\mu}\right)\\ =\ & \frac{\partial}{\partial_{P_{\alpha\beta}}}\mathrm{trace}\exp\left(\frac{1}{\mu}\begin{bmatrix}\mathbf{0} & \delta_{ij}\mathrm{I}_n\\ \delta_{ij}\mathrm{I}_n & \mathbf{0}\end{bmatrix} - \frac{1}{\mu}\sum_{\alpha,\beta}P_{\alpha\beta}\begin{bmatrix}\mathbf{0} & \mathbf{e}^n_\beta\mathbf{e}^{mT}_\alpha Q_{[j]}\\ Q^T_{[j]}\mathbf{e}^m_\alpha\mathbf{e}^{nT}_\beta & \mathbf{0}\end{bmatrix}\right)\\ =\ & -\frac{1}{\mu}\mathrm{trace}\left\{\exp\left(\frac{\mathcal{Q}_{[j]}(P)}{\mu}\right)\begin{bmatrix}\mathbf{0} & \mathbf{e}^n_\beta\mathbf{e}^{mT}_\alpha Q_{[j]}\\ Q^T_{[j]}\mathbf{e}^m_\alpha\mathbf{e}^{nT}_\beta & \mathbf{0}\end{bmatrix}\right\}\end{aligned} \tag{6.51}$$

according to Lemma 6.2.2. The partition

$$\exp\left(\frac{\mathcal{Q}_{[j]}(P)}{\mu}\right) = \begin{bmatrix}M^j_{1,1} & M^j_{1,2}\\ M^{jT}_{1,2} & M^j_{2,2}\end{bmatrix} \tag{6.52}$$

gives an explicit expression

$$\nabla_{P_{\alpha\beta}}\mathrm{trace}\exp\left(\frac{\mathcal{Q}_{[j]}(P)}{\mu}\right) = -\frac{2}{\mu}\mathbf{e}^{mT}_\alpha Q_{[j]}M^{jT}_{1,2}\mathbf{e}^n_\beta. \tag{6.53}$$

Therefore, we obtain

$$\begin{aligned}\nabla F_\mu(P) &= \sum_{j=1}^{p}\nabla\mathrm{trace}\exp\left(\frac{\mathcal{Q}_{[j]}(P)}{\mu}\right)\\ &= \sum_{j=1}^{p}\sum_{\alpha,\beta}\nabla_{P_{\alpha\beta}}\mathrm{trace}\exp\left(\frac{\mathcal{Q}_{[j]}(P)}{\mu}\right)\mathbf{e}^m_\alpha\mathbf{e}^{nT}_\beta\\ &= \sum_{j=1}^{p}\sum_{\alpha,\beta}-\frac{2}{\mu}\mathbf{e}^{mT}_\alpha Q_{[j]}M^{jT}_{1,2}\mathbf{e}^n_\beta\mathbf{e}^m_\alpha\mathbf{e}^{nT}_\beta\\ &= -\frac{2}{\mu}\left(\sum_{j=1}^{p}Q_{[j]}M^{jT}_{1,2}\right),\end{aligned} \tag{6.54}$$

which yields

$$\begin{aligned}\nabla\phi_\mu(P) &= \frac{\mu}{F_\mu(P)}\nabla F_\mu(P)\\ &= -\frac{2}{F_\mu(P)}\left(\sum_{j=1}^{p} Q_{[j]}M_{1,2}^{jT}\right). \end{aligned} \tag{6.55}$$

6.3 Computation Algorithm

6.3.1 Fixed-Point Iteration for Computing $\omega_\diamond(\cdot)$

We present a general fixed-point procedure to compute $\omega_\diamond$. Recall that the optimization problem defining $\omega_\diamond$ is as follows:

$$\omega_\diamond(Q,s) = \min_{\boldsymbol{z}} \frac{\|Q\boldsymbol{z}\|_\diamond}{\|\boldsymbol{z}\|_{\mathrm{b}\infty}} \text{ s.t. } \frac{\|\boldsymbol{z}\|_{\mathrm{b}1}}{\|\boldsymbol{z}\|_{\mathrm{b}\infty}} \leq s, \tag{6.56}$$

or equivalently,

$$\frac{1}{\omega_\diamond(Q,s)} = \max_{\boldsymbol{z}} \|\boldsymbol{z}\|_{\mathrm{b}\infty} \text{ s.t. } \|Q\boldsymbol{z}\|_\diamond \leq 1, \frac{\|\boldsymbol{z}\|_{\mathrm{b}1}}{\|\boldsymbol{z}\|_{\mathrm{b}\infty}} \leq s. \tag{6.57}$$

For any $s \in (1, s^*)$, we define a function over $[0,\infty)$ parameterized by s

$$f_s(\eta) = \max_{\boldsymbol{z}} \{\|\boldsymbol{z}\|_{\mathrm{b}\infty} : \|Q\boldsymbol{z}\|_\diamond \leq 1, \|\boldsymbol{z}\|_{\mathrm{b}1} \leq s\eta\}. \tag{6.58}$$

We basically replaced the $\|\boldsymbol{z}\|_{\mathrm{b}\infty}$ in the denominator of the fractional constraint in (6.57) with η. It turns out that the unique positive fixed point of $f_s(\eta)$ is exactly $1/\omega_\diamond(Q,s)$ as shown by the following proposition.

Proposition 6.3.1. *The function $f_s(\eta)$ has the following properties:*

1. *$f_s(\eta)$ is continuous in η;*
2. *$f_s(\eta)$ is strictly increasing in η;*

3. *$f_s(0) = 0$, $f_s(\eta) \geq s\eta > \eta$ for sufficiently small $\eta > 0$, and there exists $\rho < 1$ such that $f_s(\eta) < \rho\eta$ for sufficiently large η;*

4. *$f_s(\eta)$ has a unique positive fixed point $\eta^* = f_s(\eta^*)$ that is equal to $1/\omega_\diamond(Q, s)$;*

5. *For $\eta \in (0, \eta^*)$, we have $f_s(\eta) > \eta$; and for $\eta \in (\eta^*, \infty)$, we have $f_s(\eta) < \eta$;*

6. *For any $\epsilon > 0$, there exists $\rho_1(\epsilon) > 1$ such that $f_s(\eta) > \rho_1(\epsilon)\eta$ as long as $0 < \eta < (1-\epsilon)\eta^*$; and there exists $\rho_2(\epsilon) < 1$ such that $f_s(\eta) < \rho_2(\epsilon)\eta$ as long as $\eta > (1+\epsilon)\eta^*$.*

Proof. 1. Since in the optimization problem defining $f_s(\eta)$, the objective function $\|\boldsymbol{z}\|_{\mathrm{b}\infty}$ is continuous, and the constraint correspondence

$$\begin{aligned} C(\eta) : [0, \infty) &\twoheadrightarrow \mathbb{R}^{np} \\ \eta &\mapsto \{\boldsymbol{z} : \|Q\boldsymbol{z}\|_\diamond \leq 1, \|\boldsymbol{z}\|_{\mathrm{b}1} \leq s\eta\} \end{aligned} \tag{6.59}$$

is compact-valued and continuous (both upper and lower hemicontinuous), according to Berge's Maximum Theorem 2.2.1, the optimal value function $f_s(\eta)$ is continuous.

2. The monotone (non-strict) increasing property is obvious as increasing η enlarges the region over which the maximization is taken. We now show the strict increasing property. Suppose $0 < \eta_1 < \eta_2$, and $f_s(\eta_1)$ is achieved by $\boldsymbol{z}_1^* \neq 0$, namely, $f_s(\eta_1) = \|\boldsymbol{z}_1^*\|_{\mathrm{b}\infty}$, $\|Q\boldsymbol{z}_1^*\|_\diamond \leq 1$, and $\|\boldsymbol{z}_1^*\|_{\mathrm{b}1} \leq s\eta_1$. If $\|Q\boldsymbol{z}_1^*\|_\diamond < 1$, (this implies $\|\boldsymbol{z}_1^*\|_{\mathrm{b}1} = s\eta_1$), we define $\boldsymbol{z}_2 = c\boldsymbol{z}_1^*$ with $c = \min(1/\|Q\boldsymbol{z}_1^*\|_\diamond, s\eta_2/\|\boldsymbol{z}_1^*\|_{\mathrm{b}1}) > 1$. We then have $\|Q\boldsymbol{z}_2\|_\diamond \leq 1$, $\|\boldsymbol{z}_2\|_{\mathrm{b}1} \leq s\eta_2$, and $f_s(\eta_2) \geq \|\boldsymbol{z}_2\|_{\mathrm{b}\infty} = c\|\boldsymbol{z}_1^*\|_{\mathrm{b}\infty} > f_s(\eta_1)$.

   Consider the remaining case that $\|Q\boldsymbol{z}_1^*\|_\diamond = 1$. Without loss of generality, suppose $\|\boldsymbol{z}_1^*\|_{\mathrm{b}\infty} = \max_{1 \leq j \leq p} \|\boldsymbol{z}_{1[j]}^*\|_2$ is achieved by the block $\boldsymbol{z}_{1[1]}^* \neq 0$. Since $Q_1\boldsymbol{z}_{1[1]}^*$ is linearly dependent with the columns of $\{Q_{[j]}, j > 1\}$ (m is much less than $(p-1)n$), there exist $\{\boldsymbol{\alpha}_j \in \mathbb{R}^n\}_{j>1}$ such that $Q_{[1]}\boldsymbol{z}_{1[1]}^* + \sum_{j=2}^p Q_{[j]}\boldsymbol{\alpha}_j = 0$. Define $\boldsymbol{\alpha} = \left[\boldsymbol{z}_{1[1]}^{*T}, \boldsymbol{\alpha}_2^T, \cdots, \boldsymbol{\alpha}_p^T\right] \in \mathbb{R}^{np}$ satisfying $Q\boldsymbol{\alpha} = 0$ and $\boldsymbol{z}_2 = \boldsymbol{z}_1^* + c\boldsymbol{\alpha}$ for $c > 0$ sufficiently small such that $\|\boldsymbol{z}_2\|_{\mathrm{b}1} \leq \|\boldsymbol{z}_1^*\|_{\mathrm{b}1} + c\|\boldsymbol{\alpha}\|_{\mathrm{b}1} \leq s\eta_1 + c\|\boldsymbol{\alpha}\|_{\mathrm{b}1} \leq s\eta_2$. Clearly, $\|Q\boldsymbol{z}_2\|_\diamond = \|Q\boldsymbol{z}_1^* + cQ\boldsymbol{\alpha}\|_\diamond = \|Q\boldsymbol{z}_1^*\|_\diamond = 1$. As a consequence, we have $f_s(\eta_2) \geq \|\boldsymbol{z}_2\|_{\mathrm{b}\infty} \geq \|\boldsymbol{z}_{2[1]}\|_2 = (1+c)\|\boldsymbol{z}_{1[1]}^*\|_2 > \|\boldsymbol{z}_1^*\|_{\mathrm{b}\infty} = f_s(\eta_1)$.

The case for $\eta_1 = 0$ is proved by continuity.

3. Next we show $f_s(\eta) > s\eta$ for sufficiently small $\eta > 0$. Take $\boldsymbol{z}$ as the vector whose first element is $s\eta$ and zero otherwise. We have $\|\boldsymbol{z}\|_{\mathrm{b1}} = s\eta$ and $\|\boldsymbol{z}\|_{\mathrm{b}\infty} = s\eta > \eta$ (recall $s \in (1, \infty)$). In addition, when $\eta > 0$ is sufficiently small, we also have $\|Q\boldsymbol{z}\|_\diamond \leq 1$. Therefore, for sufficiently small η, we have $f_s(\eta) \geq s\eta > \eta$.

   We next prove the existence of $\eta_B > 0$ and $\rho_B \in (0, 1)$ such that

$$f_s(\eta) < \rho_B \eta, \ \forall \ \eta > \eta_B. \tag{6.60}$$

We use contradiction to prove this statement. Suppose for all $\eta_B > 0$ and $\rho_B \in (0, 1)$, there exists $\eta > \eta_B$ such that $f_s(\eta) \geq \rho_B \eta$. Construct sequences $\{\eta^{(k)}\}_{k=1}^\infty \subset (0, \infty)$, $\{\rho^{(k)}\}_{k=1}^\infty \subset (0, 1)$, and $\{\boldsymbol{z}^{(k)}\}_{k=1}^\infty \subset \mathbb{R}^{np}$ such that

$$\begin{aligned}
&\lim_{k\to\infty} \eta^{(k)} = \infty, \\
&\lim_{k\to\infty} \rho^{(k)} = 1, \\
&\rho^{(k)}\eta^{(k)} \leq f_s(\eta^{(k)}) = \|\boldsymbol{z}^{(k)}\|_{\mathrm{b}\infty}, \\
&\|Q\boldsymbol{z}^{(k)}\|_\diamond \leq 1, \\
&\|\boldsymbol{z}^{(k)}\|_{\mathrm{b1}} \leq s\eta^{(k)}.
\end{aligned} \tag{6.61}$$

Decompose $\boldsymbol{z}^{(k)} = \boldsymbol{z}_1^{(k)} + \boldsymbol{z}_2^{(k)}$ where $\boldsymbol{z}_1^{(k)}$ is in the null space of Q and $\boldsymbol{z}_2^{(k)}$ in the orthogonal complement of the null space of Q. The sequence $\{\boldsymbol{z}_1^{(k)}\}_{k=1}^\infty$ is bounded since $c\|z_1^{(k)}\| \leq \|Q\boldsymbol{z}_1^{(k)}\|_\diamond \leq 1$ where $c = \inf_{\boldsymbol{z}:Q\boldsymbol{z}=0} \|Q\boldsymbol{z}\|_\diamond / \|\boldsymbol{z}\| > 0$ and $\|\cdot\|$ is any norm. Then $\infty = \lim_{k\to\infty} \|\boldsymbol{z}^{(k)}\|_{\mathrm{b}\infty} \leq \lim_{k\to\infty}(\|\boldsymbol{z}_1^{(k)}\|_{\mathrm{b}\infty} + \|\boldsymbol{z}_2^{(k)}\|_{\mathrm{b}\infty})$ implies $\{\boldsymbol{z}_2^{(k)}\}_{k=1}^\infty$ is unbounded. For sufficiently large k, we proceed as follows:

$$s^* > \frac{s}{\left(\frac{s}{s^*}\right)^{1/4}} \geq \frac{s\eta^{(k)}}{\rho^{(k)}\eta^{(k)}} \geq \frac{\|\boldsymbol{z}^{(k)}\|_{\mathrm{b1}}}{\|\boldsymbol{z}^{(k)}\|_{\mathrm{b}\infty}} \geq \frac{\|\boldsymbol{z}_2^{(k)}\|_{\mathrm{b1}} - \|\boldsymbol{z}_1^{(k)}\|_{\mathrm{b1}}}{\|\boldsymbol{z}_2^{(k)}\|_{\mathrm{b}\infty} + \|\boldsymbol{z}_1^{(k)}\|_{\mathrm{b}\infty}} \geq \left(\frac{s}{s^*}\right)^{1/4} \frac{\|\boldsymbol{z}_2^{(k)}\|_{\mathrm{b1}}}{\|\boldsymbol{z}_2^{(k)}\|_{\mathrm{b}\infty}} \tag{6.62}$$

where the second and last inequalities hold only for sufficiently large k and the last inequality is due to the unboundedness of $\{\boldsymbol{z}_2^{(k)}\}$ and boundedness of $\{\boldsymbol{z}_1^{(k)}\}$. As a consequence, we have

$$\frac{\|\boldsymbol{z}_2^{(k)}\|_{\mathrm{b1}}}{\|\boldsymbol{z}_2^{(k)}\|_{\mathrm{b}\infty}} \leq \frac{s}{\sqrt{\frac{s}{s^*}}} = \sqrt{ss^*} < s^* \text{ with } \ Q\boldsymbol{z}_2^{(k)} = 0, \tag{6.63}$$

which contradicts with the definition of s^*.

4. Next we show $f_s(\eta)$ has a unique positive fixed point η^*, which is equal to $\gamma^* \overset{\text{def}}{=} 1/\omega_\diamond(Q,s)$. Properties 1) and 3) imply that there must be at least one fixed point.

   To show the uniqueness, we first prove $\gamma^* \geq \eta^*$ for any fixed point $\eta^* = f_s(\eta^*)$. Suppose $\boldsymbol{z}^*$ achieves the optimization problem defining $f_s(\eta^*)$, *i.e.*,

   $$\eta^* = f_s(\eta^*) = \|\boldsymbol{z}^*\|_{\mathrm{b}\infty}, \|Q\boldsymbol{z}^*\|_\diamond \leq 1, \|\boldsymbol{z}^*\|_{\mathrm{b}1} \leq s\eta^*. \tag{6.64}$$

   Since $\|\boldsymbol{z}^*\|_{\mathrm{b}1}/\|\boldsymbol{z}^*\|_{\mathrm{b}\infty} \leq s\eta^*/\eta^* \leq s$, we have

   $$\gamma^* \;\;\geq\;\; \frac{\|\boldsymbol{z}^*\|_{\mathrm{b}\infty}}{\|Q\boldsymbol{z}^*\|_\diamond} \geq \eta^*. \tag{6.65}$$

   If $\eta^* < \gamma^*$, we define $\eta_0 = (\eta^* + \gamma^*)/2$ and

   $$\begin{aligned} \boldsymbol{z}^{\mathrm{c}} &= \mathrm{argmax}_{\boldsymbol{z}} \frac{s\|\boldsymbol{z}\|_{\mathrm{b}\infty}}{\|\boldsymbol{z}\|_{\mathrm{b}1}} \text{ s.t. } \;\; \|Q\boldsymbol{z}\|_\diamond \leq 1, \|\boldsymbol{z}\|_{\mathrm{b}\infty} \geq \eta_0, \\ \rho &= \frac{s\|\boldsymbol{z}^{\mathrm{c}}\|_{\mathrm{b}\infty}}{\|\boldsymbol{z}^{\mathrm{c}}\|_{\mathrm{b}1}}. \end{aligned} \tag{6.66}$$

   Suppose $\boldsymbol{z}^{**}$ with $\|Q\boldsymbol{z}^{**}\|_\diamond = 1$ achieves the optimum of the optimization defining $\gamma^* = 1/\omega_\diamond(Q,s)$. Clearly, $\|\boldsymbol{z}^{**}\|_{\mathrm{b}\infty} = \gamma^* > \eta_0$, which implies $\boldsymbol{z}^{**}$ is a feasible point of the optimization problem defining $\boldsymbol{z}^{\mathrm{c}}$ and ρ. As a consequence, we have

   $$\rho \geq \frac{s\|\boldsymbol{z}^{**}\|_{\mathrm{b}\infty}}{\|\boldsymbol{z}^{**}\|_{\mathrm{b}1}} \geq 1. \tag{6.67}$$

   Actually we will show that $\rho > 1$. If $\|\boldsymbol{z}^{**}\|_{\mathrm{b}1} < s\|\boldsymbol{z}^{**}\|_{\mathrm{b}\infty}$, we are done. If not (*i.e.*, $\|\boldsymbol{z}^{**}\|_{\mathrm{b}1} = s\|\boldsymbol{z}^{**}\|_{\mathrm{b}\infty}$), as illustrated in Figure 6.1, we consider $\boldsymbol{\xi} = \frac{\eta_0}{\gamma^*}\boldsymbol{z}^{**}$, which satisfies

   $$\|Q\boldsymbol{\xi}\|_\diamond \leq \frac{\eta_0}{\gamma^*} < 1, \tag{6.68}$$

   $$\|\boldsymbol{\xi}\|_{\mathrm{b}\infty} = \eta_0, \text{ and} \tag{6.69}$$

   $$\|\boldsymbol{\xi}\|_{\mathrm{b}1} = s\eta_0. \tag{6.70}$$

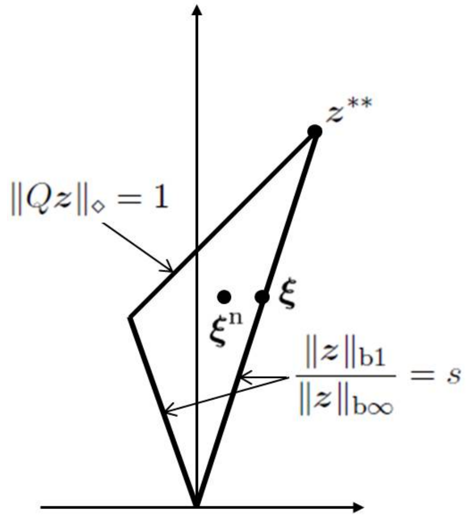


Figure 6.1: Illustration of the proof for $\rho > 1$.

To get $\boldsymbol{\xi}^{\rm n}$ as shown in Figure 6.1, pick the block of $\boldsymbol{\xi}$ with smallest non-zero ℓ_2 norm, and scale that block by a small positive constant less than 1. Because $s > 1$, $\boldsymbol{\xi}$ has more than one non-zero blocks, implying $\|\boldsymbol{\xi}^{\rm n}\|_{\rm b\infty}$ will remain the same. If the scaling constant is close enough to 1, $\|Q\boldsymbol{\xi}^{\rm n}\|_\diamond$ will remain less than 1. But the good news is that $\|\boldsymbol{\xi}^{\rm n}\|_{\rm b1}$ decreases, and hence $\rho \geq \frac{s\|\boldsymbol{\xi}^{\rm n}\|_{\rm b\infty}}{\|\boldsymbol{\xi}^{\rm n}\|_{\rm b1}}$ becomes greater than 1.

Now we proceed to obtain a contradiction that $f_s(\eta^*) > \eta^*$. If $\|\boldsymbol{z}^{\rm c}\|_{\rm b1} \leq s \cdot \eta^*$, then it is a feasible point of

$$\max_{\boldsymbol{z}} \|\boldsymbol{z}\|_{\rm b\infty} \text{ s.t. } \ \|Q\boldsymbol{z}\|_\diamond \leq 1, \|\boldsymbol{z}\|_{\rm b1} \leq s \cdot \eta^*. \tag{6.71}$$

As a consequence, $f_s(\eta^*) \geq \|\boldsymbol{z}^{\rm c}\|_{\rm b\infty} \geq \eta_0 > \eta^*$, contradicting with η^* is a fixed point and we are done. If this is not the case, *i.e.*, $\|\boldsymbol{z}^{\rm c}\|_{\rm b1} > s \cdot \eta^*$, we define a new point

$$\boldsymbol{z}^{\rm n} = \tau \boldsymbol{z}^{\rm c} \tag{6.72}$$

with

$$\tau = \frac{s \cdot \eta^*}{\|\boldsymbol{z}^{\rm c}\|_{\rm b1}} < 1. \tag{6.73}$$

Note that $\boldsymbol{z}^{\mathrm{n}}$ is a feasible point of the optimization problem defining $f_s(\eta^*)$ since

$$\|Q\boldsymbol{z}^{\mathrm{n}}\|_\diamond = \tau\|Q\boldsymbol{z}^{\mathrm{c}}\|_\diamond < 1, \text{ and} \tag{6.74}$$

$$\|\boldsymbol{z}^{\mathrm{n}}\|_{\mathrm{b}1} = \tau\|\boldsymbol{z}^{\mathrm{c}}\|_{\mathrm{b}1} = s \cdot \eta^*. \tag{6.75}$$

Furthermore, we have

$$\|\boldsymbol{z}^{\mathrm{n}}\|_{\mathrm{b}\infty} = \tau\|\boldsymbol{z}^{\mathrm{c}}\|_{\mathrm{b}\infty} = \rho\eta^*. \tag{6.76}$$

As a consequence, we obtain

$$f_s(\eta^*) \quad \geq \quad \rho\eta^* > \eta^*. \tag{6.77}$$

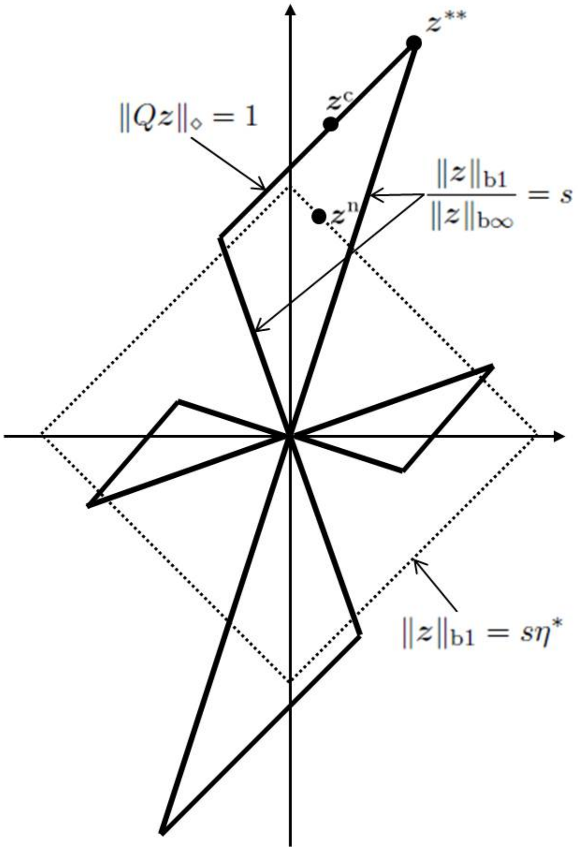

z^{**}
z^{c}
$\|Qz\|_\diamond = 1$
z^{n}
$\frac{\|z\|_{\mathrm{b}1}}{\|z\|_{\mathrm{b}\infty}} = s$
$\|z\|_{\mathrm{b}1} = s\eta^*$

Figure 6.2: Illustration of the proof for $f_s(\eta^*) \geq \rho\eta^*$.

Therefore, for any positive fixed point η^*, we have $\eta^* = \gamma^*$, *i.e.*, the positive fixed point is unique.

5. Property 5) is a consequence of 1), 3), and 4).

6. We demonstrate only the existence of $\rho_2(\epsilon)$. The existence of $\rho_1(\epsilon)$ can be proved in a similar manner, and hence is omitted.

   We need to show that for fixed $\epsilon > 0$, there exists $\rho(\epsilon) < 1$ such that for any $\eta \geq (1+\epsilon)\eta^*$ we have

   $$f_s(\eta) \leq \rho(\epsilon)\eta. \tag{6.78}$$

   In view of (6.60), we need to prove the above statement only for $\eta \in [(1+\epsilon)\eta^*, \eta_B]$. We use contradiction. Suppose for any $\rho \in (0,1)$ there exists $\eta \in [(1+\epsilon)\eta^*, \eta_B]$ such that $f_s(\eta) > \rho\eta$. Construct sequences $\{\eta^{(k)}\}_{k=1}^{\infty} \subset [(1+\epsilon)\eta^*, \eta_B]$ and $\{\rho^{(k)}\}_{k=1}^{\infty} \subset (0,1)$ with

   $$\begin{aligned} &\lim_{k\to\infty} \rho^{(k)} = 1, \\ &f_s(\eta^{(k)}) > \rho^{(k)}\eta^{(k)}. \end{aligned} \tag{6.79}$$

   Due to the compactness of $[(1+\epsilon)\eta^*, \eta_B]$, there must exist a subsequence $\{\eta^{(k_l)}\}_{l=1}^{\infty}$ of $\{\eta^{(k)}\}_{k=1}^{\infty}$ such that $\lim_{l\to\infty} \eta^{(k_l)} = \eta_{\lim}$ for some $\eta_{\lim} \in [(1+\epsilon)\eta^*, \eta_B]$. As a consequence of the continuity of $f_s(\eta)$, we have

   $$f_s(\eta_{\lim}) = \lim_{l\to\infty} f_s(\eta^{(k_l)}) \geq \lim_{l\to\infty} \rho^{(k_l)}\eta^{(k_l)} = \eta_{\lim}. \tag{6.80}$$

   Again due to the continuity of $f_s(\eta)$ and the fact that $f_s(\eta) < \eta$ for $\eta > \eta_B$, there exists $\eta_c \in [\eta_{\lim}, \eta_B]$ such that

   $$f_s(\eta_c) \quad = \quad \eta_c, \tag{6.81}$$

   contradicting with the uniqueness of the fixed point for $f_s(\eta)$.

   The result implies that starting from any initial point below the fixed point, through the iteration $\eta_{t+1} = f_s(\eta_t)$ we can approach an arbitrarily small neighborhood of the fixed point exponentially fast.

□

We have transformed the problem of computing $\omega_\diamond(Q,s)$ into one of finding the positive fixed point of a one-dimensional function $f_s(\eta)$. The property 6) of Proposition 6.3.1 states that we could start with any η_0 and use the iteration

$$\eta_{t+1} = f_s(\eta_t), t = 0, 1, \cdots \tag{6.82}$$

to find the positive fixed point η^*. In addition, if we start from two initial points, one less than η^* and one greater than η^*, then the gap between the generated sequences indicates how close we are from the fixed point η^*.

Property 5) suggests finding η^* by bisection search. Suppose we have an interval $(\eta_{\mathrm{L}}, \eta_{\mathrm{U}})$ that includes η^*. Consider the middle point $\eta_{\mathrm{M}} = \frac{\eta_{\mathrm{L}}+\eta_{\mathrm{U}}}{2}$. If $f_s(\eta_{\mathrm{M}}) < \eta_{\mathrm{M}}$, we conclude that $\eta^* < \eta_{\mathrm{M}}$ and set $\eta_{\mathrm{U}} = \eta_{\mathrm{M}}$; if $f_s(\eta_{\mathrm{M}}) > \eta_{\mathrm{M}}$, we conclude that $\eta^* > \eta_{\mathrm{M}}$ and set $\eta_{\mathrm{L}} = \eta_{\mathrm{M}}$. We continue this bisection procedure until the interval length $\eta_{\mathrm{U}} - \eta_{\mathrm{L}}$ is sufficiently small.

6.3.2 Relaxation of the Subproblem

Unfortunately, except when $n = 1$ and the signal is real, *i.e.,* the real sparse case, it is not easy to compute $f_s(\eta)$ according to (6.58). In the following theorem, we present a relaxation of the subproblem

$$\max_{\boldsymbol{z}} \|\boldsymbol{z}\|_{\mathrm{b}\infty} \text{ s.t. } \|Q\boldsymbol{z}\|_\diamond \leq 1, \|\boldsymbol{z}\|_{\mathrm{b}1} \leq s\eta \tag{6.83}$$

by computing an upper bound on $f_s(\eta)$. This proof is similar to that of Proposition 6.2.1.

Proposition 6.3.2. *When $Q = A$ and $\diamond = 2$, we have*

$$f_s(\eta) \leq \max_i \min_{P_{[i]}} \max_j s\eta \|\delta_{ij}\mathrm{I}_n - P_{[i]}^T Q_{[j]}\|_2 + \|P_{[i]}\|_2; \tag{6.84}$$

when $Q = A^T A$ *and* $\diamond = \mathrm{b}\infty$*, we have*

$$f_s(\eta) \quad \leq \quad \max_i \min_{P_{[i]}} \max_j s\eta \|\delta_{ij}\mathrm{I}_n - P_{[i]}^T Q_{[j]}\|_2 + \sum_{l=1}^p \|P_{[l],[i]}\|_2. \tag{6.85}$$

Here $P_{[i]}$ *(resp.* $Q_{[j]}$*) is the submatrix of* P *(resp.* Q*) formed by the* $(i-1)n+1$*th to* in*th columns (resp.* $(j-1)n+1$*th to* jn*th columns), and* $P_{[l],[i]}$ *is the submatrix of* P *formed by the* $(i-1)n+1$*th to* in*th columns and the* $(l-1)n+1$*th to* ln*th rows.*

For each $i = 1, \ldots, p$, the optimization problem

$$\min_{P_{[i]}} \max_j s\eta \|\delta_{ij}\mathrm{I}_n - P_{[i]}^T Q_{[j]}\|_2 + \|P_{[i]}\|_2 \tag{6.86}$$

can be solved using semidefinite programming:

$$\begin{aligned}
\min_{P_{[i]}, t_0, t_1} \quad & s\eta t_0 + t_1 \\
\text{s.t.} \quad & \|\delta_{ij}\mathrm{I}_n - P_{[i]}^T Q_{[j]}\|_2 \leq t_0, j = 1, \ldots, p; \\
& \|P_{[i]}\|_2 \leq t_1. \\
\Leftrightarrow \quad & \\
\min_{P_{[i]}, t_0, t_1} \quad & s\eta t_0 + t_1 \\
\text{s.t.} \quad & \begin{bmatrix} t_0 \mathrm{I}_n & \delta_{ij}\mathrm{I}_n - P_{[i]}^T Q_{[j]} \\ \delta_{ij}\mathrm{I}_n - Q_{[j]}^T P_{[i]} & t_0 \mathrm{I}_n \end{bmatrix} \succeq 0, j = 1, \ldots, p; \\
& \begin{bmatrix} t_1 \mathrm{I}_m & P_{[i]} \\ P_{[i]}^T & t_1 \mathrm{I}_n \end{bmatrix} \succeq 0.
\end{aligned} \tag{6.87}$$

Similarly, the optimization problem

$$\min_{P_{[i]}} \max_j s\eta \|\delta_{ij}\mathrm{I}_n - P_{[i]}^T Q_{[j]}\|_2 + \sum_{l=1}^p \|P_{[l],[i]}\|_2 \tag{6.88}$$

can be solved by the following semidefinite program:

$$
\begin{aligned}
\min_{P_{[i]},t_0,t_1,\ldots,t_p} \quad & s\eta t_0 + \sum_{l=1}^{p} t_l \\
& \text{s.t.} \quad \|\delta_{ij}\mathrm{I}_n - P_{[i]}^T Q_{[j]}\|_2 \le t_0, j = 1,\ldots,p; \\
& \|P_{[l],[i]}\|_2 \le t_l, l = 1,\ldots,p. \\
\Leftrightarrow & \\
\min_{P_{[i]},t_0,t_1} \quad & s\eta t_0 + \sum_{l=1}^{p} t_l \\
\text{s.t.} \quad & \begin{bmatrix} t_0\mathrm{I}_n & \delta_{ij}\mathrm{I}_n - P_{[i]}^T Q_{[j]} \\ \delta_{ij}\mathrm{I}_n - Q_{[j]}^T P_{[i]} & t_0\mathrm{I}_n \end{bmatrix} \succeq 0, j = 1,\ldots,p; \\
& \begin{bmatrix} t_l\mathrm{I}_n & P_{[l],[i]} \\ P_{[i]}^{lT} & t_l\mathrm{I}_n \end{bmatrix} \succeq 0, l = 1,\ldots,p. \qquad (6.89)
\end{aligned}
$$

First-order implementations using smoothing approximation are detailed in Sections 6.3.3 and 6.3.4.

6.3.3 Smoothing Technique for Solving (6.86)

We now proceed to solve the following optimization problem for a particular i:

$$\min_P \max_j \eta\|\delta_{ij}\mathrm{I}_n - P^T Q_{[j]}\|_2 + \|P\|_2, \qquad (6.90)$$

which is equivalent to

$$\min_P \eta\lambda_{\max}(\mathcal{Q}(P)) + \lambda_{\max}(\mathcal{P}(P)) \qquad (6.91)$$

with

$$\mathcal{P}(P) \quad = \quad \begin{bmatrix} \mathbf{0} & P^T \\ P & \mathbf{0} \end{bmatrix}. \qquad (6.92)$$

For any $\mu_1, \mu_2 > 0$, define the following smooth approximations

$$\begin{aligned}
\phi_{\mu_1}(P) &\overset{\text{def}}{=} \mu_1 \log\left(\operatorname{trace}\exp\left(\frac{\mathcal{Q}(P)}{\mu_1}\right)\right) \overset{\text{def}}{=} \mu_1 \log F_{\mu_1}(P), \\
\psi_{\mu_2}(P) &\overset{\text{def}}{=} \mu_2 \log\left(\operatorname{trace}\exp\left(\frac{\mathcal{P}(P)}{\mu_2}\right)\right) \overset{\text{def}}{=} \mu_2 \log G_{\mu_2}(P), \\
f_{\mu_1,\mu_2}(P) &\overset{\text{def}}{=} \eta\phi_{\mu_1}(P) + \psi_{\mu_2}(P). \qquad (6.93)
\end{aligned}$$

Similar arguments lead to the choices

$$\mu_1 = \frac{\epsilon}{4\eta\log(2np)}, \qquad (6.94)$$

$$\mu_2 = \frac{\epsilon}{4\log(m+n)}, \qquad (6.95)$$

with approximation accuracy

$$\eta\phi_{\mu_1}(P) + \psi_{\mu_2}(P) \le \eta\lambda_{\max}(\mathcal{Q}(P)) + \lambda_{\max}(\mathcal{P}(P)) \le \eta\phi_{\mu_1}(P) + \psi_{\mu_2}(P) + \frac{\epsilon}{2}. \quad (6.96)$$

The corresponding Lipschitz constants for the gradient functions of $\phi_{\mu_1}(P)$ and $\psi_{\mu_2}(P)$ are respectively

$$L_1 = \frac{4\log(2np)\max_j \|Q_{[j]}\|_2^2}{\epsilon} \qquad (6.97)$$

$$L_2 = \frac{4\log(m+n)}{\epsilon}. \qquad (6.98)$$

As a consequence, the Lipschitz constant for the gradient function of $f_{\mu_1,\mu_2}(P)$ is

$$L = \frac{4\eta\log(2np)\max_j \|Q_{[j]}\|_2^2 + 4\log(m+n)}{\epsilon}. \qquad (6.99)$$

The gradient function is expressed as

$$\nabla f_{\mu_1,\mu_2}(P) = \eta\nabla\phi_{\mu_1}(P) + \nabla\psi_{\mu_2}(P) \qquad (6.100)$$

with $\nabla\phi_{\mu_1}(P)$ given in (6.55) and

$$\nabla\psi_{\mu_2}(P) = \frac{2}{G_{\mu_2}(P)} M^{12T}, \qquad (6.101)$$

where the partition

$$\exp\left(\frac{\mathcal{P}(P)}{\mu}\right) = \begin{bmatrix} M^{11} & M^{12} \\ M^{12T} & M^{22} \end{bmatrix}. \tag{6.102}$$

6.3.4 Smoothing Technique for Solving (6.88)

The third optimization can be solved in a similar manner

$$\begin{aligned} & \min_P \max_j \eta \|\delta_{ij} \mathrm{I}_n - P^T Q_{[j]}\|_2 + \sum_{l=1}^{p} \|(P^T)^T_{[l]}\|_2 \\ & \Leftrightarrow \\ & \min_P \eta \lambda_{\max}(\mathcal{Q}(P)) + \sum_{l=1}^{p} \lambda_{\max}(\mathcal{P}((P^T)^T_{[l]})). \end{aligned} \tag{6.103}$$

Define

$$\begin{aligned} \varphi_{\mu_3}(P) &= \sum_{l=1}^{p} \mu_3 \log\left(\operatorname{trace}\exp\left(\frac{\mathcal{P}((P^T)^T_{[l]})}{\mu_3}\right)\right), \\ f_{\mu_1,\mu_3}(P) &= \eta \phi_{\mu_1}(P) + \varphi_{\mu_3}(P) \end{aligned} \tag{6.104}$$

with μ_1 given in (6.94) and $\mu_3 = \frac{\epsilon}{4p\log(np+n)}$. The Lipschitz constant for the gradient function of $f_{\mu_1,\mu_3}(P)$ is

$$L = \frac{4\eta \log(2np) \max_j \|Q_{[j]}\|_2^2 + 4p\log(np+n)}{\epsilon}. \tag{6.105}$$

The gradient of $\varphi_{\mu_3}(P)$ is

$$\nabla \varphi_{\mu_3}(P) = \begin{bmatrix} \nabla_{(P^T)^T_{[1]}} \psi_{\mu_3}((P^T)^T_{[1]}) & \cdots & \nabla_{(P^T)^T_{[p]}} \psi_{\mu_3}((P^T)^T_{[p]}) \end{bmatrix}, \tag{6.106}$$

which together with (6.55) gives the gradient of $f_{\mu_1,\mu_3}(P)$.

6.3.5 Fixed-Point Iteration for Computing a Lower Bound on $\omega_\diamond$

Although Proposition 6.3.2 provides ways to efficiently compute upper bounds on the subproblem (6.83) for fixed η, it is not obvious whether we could use it to compute an upper on the positive fixed point of $f_s(\eta)$, or $1/\omega_\diamond(Q, s)$. We show in this subsection that another iterative procedure can compute such upper bounds.

To this end, we define functions $g_{s,i}(\eta)$ and $g_s(\eta)$ over $[0, \infty)$ parameterized by s for $s \in (1, s_*)$,

$$\begin{aligned} g_{s,i}(\eta) &= \min_{P_{[i]}} s\eta \left(\max_j \|\delta_{ij} \mathrm{I}_n - P_{[i]}^T Q_{[j]}\|_2 \right) + \|P_{[i]}\|_2, \\ g_s(\eta) &= \max_i g_{s,i}(\eta). \end{aligned} \tag{6.107}$$

The following proposition lists some properties of $g_{s,i}(\eta)$ and $g_s(\eta)$.

Proposition 6.3.3. *The functions $g_{s,i}(\eta)$ and $g_s(\eta)$ have the following properties:*

1. *$g_{s,i}(\eta)$ and $g_s(\eta)$ are continuous in η;*
2. *$g_{s,i}(\eta)$ and $g_s(\eta)$ are strictly increasing in η;*
3. *$g_{s,i}(\eta)$ is concave for every i;*
4. *$g_s(0) = 0$, $g_s(\eta) \geq s\eta > \eta$ for sufficiently small $\eta > 0$, and there exists $\rho < 1$ such that $g_s(\eta) < \rho\eta$ for sufficiently large η; the same holds for $g_{s,i}(\eta)$;*
5. *$g_{s,i}$ and $g_s(\eta)$ have unique positive fixed points $\eta_i^* = g_{s,i}(\eta_i^*)$ and $\eta^* = g_s(\eta^*)$, respectively; and $\eta^* = \max_i \eta_i^*$;*
6. *For $\eta \in (0, \eta^*)$, we have $g_s(\eta) > \eta$; and for $\eta \in (\eta^*, \infty)$, we have $g_s(\eta) < \eta$; the same statement holds also for $g_{s,i}(\eta)$.*
7. *For any $\epsilon > 0$, there exists $\rho_1(\epsilon) > 1$ such that $g_s(\eta) > \rho_1(\epsilon)\eta$ as long as $0 < \eta \leq (1 - \epsilon)\eta^*$; and there exists $\rho_2(\epsilon) < 1$ such that $g_s(\eta) < \rho_2(\epsilon)\eta$ as long as $\eta > (1 + \epsilon)\eta^*$.*

Proof. 1. First note that adding an additional constraint $\|P_{[i]}\|_2 \leq s\eta$ in the definition of $g_{s,i}$ does not change the definition, because $g_{s,i}(\eta) \leq s\eta$ as easily seen by setting $P_{[i]} = 0$:

$$\begin{aligned} g_{s,i} &= \min_{P_{[i]}} \left\{ s\eta \left(\max_j \|\delta_{ij}\mathrm{I}_n - P_{[i]}^T Q_{[j]}\|_2 \right) + \|P_{[i]}\|_2 : \|P_{[i]}\|_2 \leq s\eta \right\} \\ &= -\max_{P_{[i]}} \left\{ -s\eta \left(\max_j \|\delta_{ij}\mathrm{I}_n - P_{[i]}^T Q_{[j]}\|_2 \right) - \|P_{[i]}\|_2 : \|P_{[i]}\|_2 \leq s\eta \right\}. \end{aligned} \tag{6.108}$$

Since the objective function to be maximized is continuous, and the constraint correspondence

$$C(\eta) = \left\{ P_{[i]} : \|P_{[i]}\|_2 \leq s\eta \right\} \tag{6.109}$$

is compact-valued and continuous (both upper and lower hemicontinuous), according to Berge's Maximum Theorem [44], the optimal value function $g_{s,i}(\eta)$ is continuous. The continuity of $g_s(\eta)$ follows from that finite maximization preserves the continuity.

2. To show the strict increasing property, suppose $\eta_1 < \eta_2$ and $P_{[i]}^2$ achieves $g_{s,i}(\eta_2)$. Then we have

$$\begin{aligned} g_{s,i}(\eta_1) &\leq s\eta_1 \left(\max_j \|\delta_{ij}\mathrm{I}_n - P_{[i]}^{2T} Q_{[j]}\|_2 \right) + \|P_{[i]}^2\|_2 \\ &< s\eta_2 \left(\max_j \|\delta_{ij}\mathrm{I}_n - P_{[i]}^{2T} Q_{[j]}\|_2 \right) + \|P_{[i]}^2\|_2 \\ &= g_{s,i}(\eta_2). \end{aligned} \tag{6.110}$$

The strict increasing of $g_s(\eta)$ then follows immediately.

3. The concavity of $g_{s,i}(\eta)$ follows from the fact that $g_{s,i}(\eta)$ is the minimization of a function of variables η and $P_{[i]}$, and when $P_{[i]}$, the variable to be minimized, is fixed, the function is linear in η.

4. Next we show that when $\eta > 0$ is sufficiently small $g_s(\eta) \geq s\eta$. For any i, we have the following,

$$\begin{aligned} g_{s,i}(\eta) &= \min_{P_{[i]}} s\eta \left(\max_j \|\delta_{ij}\mathrm{I}_n - P_{[i]}^T Q_{[j]}\|_2 \right) + \|P_{[i]}\|_2 \\ &\geq \min_{P_{[i]}} s\eta \left(1 - \|P_{[i]}^T Q_{[i]}\|_2\right) + \|P_{[i]}\|_2 \\ &\geq \min_{P_{[i]}} s\eta \left(1 - \|P_{[i]}\|_2 \|Q_{[i]}\|_2\right) + \|P_{[i]}\|_2 \\ &= s\eta + \min_{P_{[i]}} \|P_{[i]}\|_2 \left(1 - s\eta\|Q_{[i]}\|_2\right) \\ &\geq s\eta > \eta, \end{aligned} \tag{6.111}$$

where the minimum of the last optimization problem is achieved at $P_{[i]} = 0$ when $\eta < 1/(s\|Q_{[i]}\|_2)$. Clearly, $g_s(\eta) = \max_i g_{s,i}(\eta) \geq s\eta > \eta$ for such η.

Recall that

$$\frac{1}{s_*} = \max_i \min_{P_{[i]}} \max_j \|\delta_{ij}\mathrm{I}_n - P_{[i]}^T Q_{[j]}\|_2. \tag{6.112}$$

Suppose $P_{[i]}^*$ is the optimal solution for each $\min_{P_{[i]}} \max_j \|\delta_{ij}\mathrm{I}_n - P_{[i]}^T Q_{[j]}\|_2$. For each i, we then have

$$\frac{1}{s_*} \geq \max_j \|\delta_{ij}\mathrm{I}_n - P_{[i]}^{*T} Q_{[j]}\|_2, \tag{6.113}$$

which implies

$$\begin{aligned} g_{s,i}(\eta) &= \min_{P_{[i]}} s\eta \left(\max_j \|\delta_{ij}\mathrm{I}_n - P_{[i]}^T Q_{[j]}\|_2 \right) + \|P_{[i]}\|_2 \\ &\leq s\eta \left(\max_j \|\delta_{ij}\mathrm{I}_n - P_{[i]}^{*T} Q_{[j]}\|_2 \right) + \|P_{[i]}^*\|_2 \\ &\leq \frac{s}{s_*}\eta + \|P_{[i]}^*\|_2. \end{aligned} \tag{6.114}$$

As a consequence, we obtain

$$g_s(\eta) = \max_i g_{s,i}(\eta) \leq \frac{s}{s_*}\eta + \max_i \|P_{[i]}^*\|_2. \tag{6.115}$$

Pick $\rho \in (s/s_*, 1)$. Then, we have the following when $\eta > \max_i \|P^*_{[i]}\|_2/(\rho - s/s_*)$:

$$g_s(\eta) \leq \rho\eta. \tag{6.116}$$

5. We first show the existence and uniqueness of the positive fixed points for $g_{s,i}(\eta)$. The properties 1) and 4) imply that $g_{s,i}(\eta)$ has at least one positive fixed point. To prove uniqueness, suppose there are two fixed points $0 < \eta_1^* < \eta_2^*$. Pick η_0 small enough such that $g_{s,i}(\eta_0) > \eta_0 > 0$ and $\eta_0 < \eta_1^*$. Then $\eta_1^* = \lambda\eta_0 + (1-\lambda)\eta_2^*$ for some $\lambda \in (0,1)$, which implies that $g_{s,i}(\eta_1^*) \geq \lambda g_{s,i}(\eta_0) + (1-\lambda)g_{s,i}(\eta_2^*) > \lambda\eta_0 + (1-\lambda)\eta_2^* = \eta_1^*$ due to the concavity, contradicting with $\eta_1^* = g_{s,i}(\eta_1^*)$.

   The set of positive fixed point for $g_s(\eta)$, $\{\eta \in (0,\infty) : \eta = g_s(\eta) = \max_i g_{s,i}(\eta)\}$, is a subset of $\bigcup_{i=1}^p \{\eta \in (0,\infty) : \eta = g_{s,i}(\eta)\} = \{\eta_i^*\}_{i=1}^p$. We argue that

   $$\eta^* = \max_i \eta_i^* \tag{6.117}$$

   is the unique positive fixed point for $g_s(\eta)$.

   We proceed to show that η^* is a fixed point of $g_s(\eta)$. Suppose η^* is a fixed point of $g_{s,i_0}(\eta)$, then it suffices to show that $g_s(\eta^*) = \max_i g_{s,i}(\eta^*) = g_{s,i_0}(\eta^*)$. If this is not the case, there exists $i_1 \neq i_0$ such that $g_{s,i_1}(\eta^*) > g_{s,i_0}(\eta^*) = \eta^*$. The continuity of $g_{s,i_1}(\eta)$ and the property 4) imply that there exists $\eta > \eta^*$ with $g_{s,i_1}(\eta) = \eta$, contradicting with the definition of η^*.

   To show the uniqueness, suppose η_1^* is fixed point of $g_{s,i_1}(\eta)$ satisfying $\eta_1^* < \eta^*$. Then, we must have $g_{s,i_0}(\eta_1^*) > g_{s,i_1}(\eta_1^*)$ because otherwise the continuity implies the existence of another fixed point of $g_{s,i_0}(\eta)$. As a consequence, $g_s(\eta_1^*) > g_{s,i_1}(\eta_1^*) = \eta_1^*$ and η_1^* is not a fixed point of $g_s(\eta)$.

6. This property simply follows from the continuity, the uniqueness, and property 4).

7. We use contradiction to show the existence of $\rho_1(\epsilon)$ in 7). In view of 4), we need only to show the existence of such a $\rho_1(\epsilon)$ that works for $\eta_L \leq \eta \leq (1-\epsilon)\eta^*$ where $\eta_L = \sup\{\eta : g_s(\xi) > s\xi, \forall 0 < \xi \leq \eta\}$. Suppose otherwise, we then

construct sequences $\{\eta^{(k)}\}_{k=1}^{\infty} \subset [\eta_L, (1-\epsilon)\eta^*]$ and $\{\rho_1^{(k)}\}_{k=1}^{\infty} \subset (1, \infty)$ with

$$\begin{aligned} &\lim_{k\to\infty} \rho_1^{(k)} = 1, \\ &g_s(\eta^{(k)}) \leq \rho^{(k)}\eta^{(k)}. \end{aligned} \tag{6.118}$$

Due to the compactness of $[\eta_L, (1-\epsilon)\eta^*]$, there must exist a subsequence $\{\eta^{(k_l)}\}_{l=1}^{\infty}$ of $\{\eta^{(k)}\}$ such that $\lim_{l\to\infty} \eta^{(k_l)} = \eta_{\lim}$ for some $\eta_{\lim} \in [\eta_L, (1-\epsilon)\eta^*]$. As a consequence of the continuity of $g_s(\eta)$, we have

$$g_s(\eta_{\lim}) = \lim_{l\to\infty} g_s(\eta^{(k_l)}) \leq \lim_{l\to\infty} \rho_1^{(k_l)}\eta^{(k_l)} = \eta_{\lim}. \tag{6.119}$$

Again due to the continuity of $g_s(\eta)$ and the fact that $g_s(\eta) < \eta$ for $\eta < \eta_L$, there exists $\eta_c \in [\eta_L, \eta_{\lim}]$ such that

$$g_s(\eta_c) \quad = \quad \eta_c, \tag{6.120}$$

contradicting with the uniqueness of the fixed point for $g_s(\eta)$. The existence of $\rho_2(\epsilon)$ can be proved in a similar manner.

□

The same properties in Proposition 6.3.3 hold for the functions defined below:

$$\begin{aligned} h_{s,i}(\eta) &= \min_{P_{[i]}} s\eta \left(\max_j \|\delta_{ij}\mathrm{I}_n - P_{[i]}^T Q_{[j]}\|_2 \right) + \sum_{l=1}^{p} \|P_{[l],[i]}\|_2, && (6.121) \\ h_s(\eta) &= \max_i h_{s,i}(\eta). && (6.122) \end{aligned}$$

Proposition 6.3.4. *The functions $h_{s,i}(\eta)$ and $h_s(\eta)$ have the following properties:*

1. *$h_{s,i}(\eta)$ and $h_s(\eta)$ are continuous in η;*
2. *$h_{s,i}(\eta)$ and $h_s(\eta)$ are strictly increasing in η;*
3. *$h_{s,i}(\eta)$ is concave for every i;*

4. *$h_s(0) = 0$, $h_s(\eta) \geq s\eta > \eta$ for sufficiently small $\eta > 0$, and there exists $\rho < 1$ such that $h_s(\eta) < \rho\eta$ for sufficiently large η; the same holds for $h_{s,i}(\eta)$;*

5. *$h_{s,i}$ and $h_s(\eta)$ have unique positive fixed points $\eta_i^* = h_{s,i}(\eta_i^*)$ and $\eta^* = h_s(\eta^*)$, respectively; and $\eta^* = \max_i \eta_i^*$;*

6. *For $\eta \in (0, \eta^*)$, we have $h_s(\eta) > \eta$; and for $\eta \in (\eta^*, \infty)$, we have $h_s(\eta) < \eta$; the same statement holds also for $h_{s,i}(\eta)$.*

7. *For any $\epsilon > 0$, there exists $\rho_1(\epsilon) > 1$ such that $h_s(\eta) > \rho_1(\epsilon)\eta$ as long as $0 < \eta \leq (1-\epsilon)\eta^*$; and there exists $\rho_2(\epsilon) < 1$ such that $h_s(\eta) < \rho_2(\epsilon)\eta$ as long as $\eta > (1+\epsilon)\eta^*$.*

An immediate consequence of Propositions 6.3.3 and 6.3.4 is the following:

Theorem 6.3.1. *Suppose η^* is the unique fixed point of $g_s(\eta)$ ($h_s(\eta)$, resp.), then we have*

$$\eta^* \geq \frac{1}{\omega_2(A,s)} \left(\frac{1}{\omega_{\mathrm{b}\infty}(A^TA,s)}, \text{ resp.} \right). \tag{6.123}$$

Proposition 6.3.3 implies three ways to compute the fixed point η^* for $g_s(\eta)$. The same discussion is also valid for $h_s(\eta)$.

1. **Naive Fixed-Point Iteration:** Property 7) of Proposition 6.3.3 suggests that the fixed point iteration

$$\eta_{t+1} = g_s(\eta_t), t = 0, 1, \ldots \tag{6.124}$$

starting from any initial point η_0 converges to η^*, no matter $\eta_0 < \eta^*$ or $\eta_0 > \eta^*$. The algorithm can be made more efficient in the case $\eta_0 < \eta^*$. More specifically, since $g_s(\eta) = \max_i g_{s,i}(\eta)$, at each fixed-point iteration, we set η_{t+1} to be the first $g_{s,i}(\eta_t)$ that is greater than $\eta_t + \epsilon$ with ϵ some tolerance parameter. If for all i, $g_{s,i}(\eta_t) < \eta_t + \epsilon$, then $g_s(\eta_t) = \max_i g_{s,i}(\eta_t) < \eta_t + \epsilon$, which indicates the optimal function value can not be improved greatly and the algorithm should terminate. In most cases, to get η_{t+1}, we need to solve only one optimization problem $\min_{P_{[i]}} s\eta \left(\max_j \|\delta_{ij}\mathrm{I}_n - P_{[i]}^T Q_{[j]}\|_2 \right) + \|P_{[i]}\|_2$ instead of p. This is in contrast to the case where $\eta_0 > \eta^*$, because in the later case we must compute all $g_{s,i}(\eta_t)$ to update $\eta_{t+1} = \max_i g_{s,i}(\eta_t)$. An update based on a single $g_{s,i}(\eta_t)$ might generate a value smaller than η^*.

The naive fixed-point iteration has two major disadvantages. Firstly, the stopping criterion based on successive improvement is not accurate as it does not reflect the gap between η_t and η^*. This disadvantage can be remedied by starting from both below and above η^*. The distance between corresponding terms in the two generated sequences is an indication of the gap to the fixed point η^*. However, the resulting algorithm is very slow, especially when updating η_{t+1} from above η^*. Secondly, the iteration process is slow, especially when close to the fixed point η^*. This is because $\rho_1(\epsilon)$ and $\rho_2(\epsilon)$ in 7) of Proposition 6.3.3 are close to 1.

2. **Bisection:** The bisection approach is motivated by property 6) of Proposition 6.3.3. Starting from an initial interval $(\eta_{\rm L}, \eta_{\rm U})$ that contains η^*, we compute $g_s(\eta_{\rm M})$ with $\eta_{\rm M} = (\eta_{\rm L} + \eta_{\rm U})/2$. As a consequence of property 6), $g_s(\eta_{\rm M}) > \eta_{\rm M}$ implies $g_s(\eta_{\rm M}) < \eta^*$, and we set $\eta_{\rm L} = g_s(\eta_{\rm M})$; $g_s(\eta_{\rm M}) < \eta_{\rm M}$ implies $g_s(\eta_{\rm M}) > \eta^*$, and we set $\eta_{\rm U} = g_s(\eta_{\rm M})$. The bisection process can also be accelerated by setting $\eta_{\rm L} = g_{s,i}(\eta_{\rm M})$ for the first $g_{s,i}(\eta_{\rm M})$ greater than $\eta_{\rm M}$. The convergence of the bisection approach is much faster than the naive fixed point iteration because each iteration reduces the interval length at least by half. In addition, half the length of the interval is an upper bound on the gap between $\eta_{\rm M}$ and η^*, resulting an accurate stopping criterion. However, if the initial $\eta_{\rm U}$ is too larger than η^*, the majority of $g_s(\eta_{\rm M})$ would turn out to be less than η^*. The verification of $g_s(\eta_{\rm M}) < \eta_{\rm M}$ needs solving p semidefinite programs, greatly degrading the algorithm's performance.

3. **Fixed-Point Iteration + Bisection:** The third approach combines the advantages of the bisection method and the fixed-point iteration method, at the level of $g_{s,i}(\eta)$. This method relies heavily on the representation $g_s(\eta) = \max_i g_{s,i}(\eta)$ and $\eta^* = \max_i \eta_i^*$.

   Starting from an initial interval $(\eta_{\rm L0}, \eta_{\rm U})$ and the index set $\mathcal{I}_0 = \{1, \dots, p\}$, we pick any $i_0 \in \mathcal{I}_0$ and use the (accelerated) bisection method with starting interval $(\eta_{\rm L0}, \eta_{\rm U})$ to find the positive fixed point $\eta_{i_0}^*$ of $g_{s,i_0}(\eta)$. For any $i \in \mathcal{I}_0/i_0$, $g_{s,i}(\eta_{i_0}^*) \leq \eta_{i_0}^*$ implies that the fixed point η_i^* of $g_{s,i}(\eta)$ is less than or equal to

$\eta^*_{i_0}$ according to the continuity of $g_{s,i}(\eta)$ and the uniqueness of its positive fixed point. As a consequence, we remove this i from the index set $\mathcal{I}_0$. We denote $\mathcal{I}_1$ as the index set after all such is removed, *i.e.,* $\mathcal{I}_1 = \mathcal{I}_0/\{i : g_{s,i}(\eta^*_{i_0}) \leq \eta^*_{i_0}\}$. We also set $\eta_{L1} = \eta^*_{i_0}$ as $\eta^* \geq \eta^*_{i_0}$. Next we test the $i_1 \in \mathcal{I}_1$ with the *largest* $g_{s,i}(\eta^*_{i_0})$ and construct $\mathcal{I}_2$ and $\eta_{\mathrm{L}2}$ in a similar manner. We repeat the process until the index set $\mathcal{I}_t$ is empty. The η^*_i found at the last step is the maximal η^*_i, which is equal to η^*.

6.4 Probabilistic Analysis

In this section, we analyze how good are the performance bounds in Theorem 6.1.1 for random sensing matrices. For this purpose, we define the block ℓ_1-constrained minimal singular value (block ℓ_1-CMSV), which is an extension of the ℓ_1-CMSV define in Definition 4.4.1 in the sparse setting:

Definition 6.4.1. *For any $s \in [1, p]$ and matrix $A \in \mathbb{R}^{m\times np}$, define the block ℓ_1-constrained minimal singular value (abbreviated as block ℓ_1-CMSV) of A by*

$$\rho_s(A) = \min_{\boldsymbol{z}:\ \|\boldsymbol{z}\|^2_{\mathrm{b}1}/\|\boldsymbol{z}\|^2_2 \leq s} \frac{\|A\boldsymbol{z}\|_2}{\|\boldsymbol{z}\|_2}. \tag{6.125}$$

The most important difference between $\rho_s(A)$ and $\omega_\diamond(Q, s)$ is the replacement of $\|\cdot\|_{\mathrm{b}\infty}$ with $\|\cdot\|_2$ in the denominators of the fractional constraint and the objective function. The Euclidean norm $\|\cdot\|_2$ is more amenable to probabilistic analysis. The connections between $\rho_s(A)$, $\omega_2(A, s)$, and $\omega_{\mathrm{b}\infty}(A^TA, s)$, established in the following lemma, allow us to analyze the probabilistic behaviors of $\omega_\diamond(Q, s)$ using the results for $\rho_s(A)$ we are going to establish later.

Lemma 6.4.1.

$$\sqrt{s}\sqrt{\omega_{\mathrm{b}\infty}(A^TA, s)} \geq \omega_2(A, s) \geq \rho_{s^2}(A). \tag{6.126}$$

Proof. For any $\boldsymbol{z}$ such that $\|\boldsymbol{z}\|_{\mathrm{b}\infty} = 1$ and $\|\boldsymbol{z}\|_{\mathrm{b}1} \leq s$, we have

$$\begin{aligned} \boldsymbol{z}A^T A\boldsymbol{z} &= \left\langle \boldsymbol{z}, A^T A\boldsymbol{z} \right\rangle \\ &\leq \|\boldsymbol{z}\|_{\mathrm{b}1} \|A^T A\boldsymbol{z}\|_{\mathrm{b}\infty} \\ &\leq s\|A^T A\boldsymbol{z}\|_{\mathrm{b}\infty}. \end{aligned} \tag{6.127}$$

Taking the minima of both sides of (6.127) over $\{\boldsymbol{z} : \|\boldsymbol{z}\|_{\mathrm{b}\infty} = 1, \|\boldsymbol{z}\|_1 \leq s\}$ yields

$$\omega_2^2(A, s) \leq s\omega_{\mathrm{b}\infty}(A^T A, s). \tag{6.128}$$

For the other inequality, note that $\|\boldsymbol{z}\|_{\mathrm{b}1}/\|\boldsymbol{z}\|_{\mathrm{b}\infty} \leq s$ implies $\|\boldsymbol{z}\|_{\mathrm{b}1} \leq s\|\boldsymbol{z}\|_{\mathrm{b}\infty} \leq s\|\boldsymbol{z}\|_2$, or equivalently,

$$\{\boldsymbol{z} : \|\boldsymbol{z}\|_{\mathrm{b}1}/\|\boldsymbol{z}\|_{\mathrm{b}\infty} \leq s\} \subseteqq \{\boldsymbol{z} : \|\boldsymbol{z}\|_{\mathrm{b}1}/\|\boldsymbol{z}\|_2 \leq s\}. \tag{6.129}$$

As a consequence, we have

$$\begin{aligned} \omega_2(A, s) &= \min_{\|\boldsymbol{z}\|_{\mathrm{b}1}/\|\boldsymbol{z}\|_{\mathrm{b}\infty} \leq s} \frac{\|A\boldsymbol{z}\|_2}{\|\boldsymbol{z}\|_2} \frac{\|\boldsymbol{z}\|_2}{\|\boldsymbol{z}\|_{\mathrm{b}\infty}} \\ &\geq \min_{\|\boldsymbol{z}\|_{\mathrm{b}1}/\|\boldsymbol{z}\|_{\mathrm{b}\infty} \leq s} \frac{\|A\boldsymbol{z}\|_2}{\|\boldsymbol{z}\|_2} \\ &\geq \min_{\|\boldsymbol{z}\|_{\mathrm{b}1}/\|\boldsymbol{z}\|_2 \leq s} \frac{\|A\boldsymbol{z}\|_2}{\|\boldsymbol{z}\|_2} \\ &= \rho_{s^2}(A), \end{aligned} \tag{6.130}$$

where the first inequality is due to $\|\boldsymbol{z}\|_2 \geq \|\boldsymbol{z}\|_{\mathrm{b}\infty}$, and the second inequality is because the minimization is taken over a larger set. □

Next we derive a condition on the number of measurements to get $\rho_s(A)$ bounded away from zero with high probability for sensing matrices with *i.i.d.* subgaussian and isotropic rows.

Theorem 6.4.1. *Let the rows of the sensing matrix* $\sqrt{m}A$ *be* i.i.d. *subgaussian and isotropic random vectors with constant* L*. Then there exist constants* c_1 *and* c_2

depending on L such that for any $\epsilon > 0$ and $m \geq 1$ satisfying

$$m \geq c_1 \frac{sn + s\log p}{\epsilon^2}, \tag{6.131}$$

we have

$$\mathbb{E}|1 - \rho_s(A)| \leq \epsilon, \tag{6.132}$$

and

$$\mathbb{P}\{1 - \epsilon \leq \rho_s(A) \leq 1 + \epsilon\} \geq 1 - \exp(-c_2\epsilon^2 m). \tag{6.133}$$

Proof of Theorem 6.4.1. We apply Theorem 2.3.2 to estimate the block ℓ_1-CMSV. Denote $\mathcal{H}_s^n = \{\boldsymbol{u} \in \mathbb{R}^{np} : \|\boldsymbol{u}\|_2^2 = 1, \|\boldsymbol{u}\|_{\mathrm{b1}}^2 \leq s\}$, a subset of the unit sphere of $\mathbb{R}^{np}$. Similar to the proof of Theorem 4.4.1 for the sparse case, the proof of Theorem 6.4.1 boils down to computing an upper bound of $\ell_*(\mathcal{H}_s^n)$:

$$\begin{aligned} \ell_*(\mathcal{H}_s^n) &= \mathbb{E}\sup_{\boldsymbol{u}\in\mathcal{H}_s^n} \langle \boldsymbol{g}, \boldsymbol{u}\rangle \\ &\leq \mathbb{E}\|\boldsymbol{u}\|_{\mathrm{b1}}\|\boldsymbol{g}\|_{\mathrm{b}\infty} \\ &\leq \sqrt{sn} + \sqrt{s\log p} \end{aligned} \tag{6.134}$$

according to (2.69). The conclusion of Theorem 6.4.1 then follows. □

Using $\rho_s(A)$, we could equally develop bounds similar to those of Theorem 6.1.1 on the ℓ_2 norm of the error vectors. For example, the error bound for the BS-BP would look like

$$\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2 \leq \frac{2\varepsilon}{\rho_{2k}(A)}. \tag{6.135}$$

The conclusion of Theorem 6.4.1 combined with the previous equation implies that we could stably recover a block sparse signal using BS-BP with high probability if the sensing matrix is subgaussian and isotropic and $m \geq c(kn + k\log p)/\epsilon^2$. If we do not consider the block structure in the signal, we would need $m \geq c(kn\log p)/\epsilon^2$ measurements as the sparsity level is kn [87]. Therefore, the prior information regarding the block structure greatly reduces the number of measurements necessary to recover the signal. The lower bound on m is essentially the same as the one given by the block RIP (See [6, Proposition 4] and the end of Section 5.4).

Theorem 6.4.2. *Under the assumptions and notations of Theorem 6.4.1, there exist constants* c_1, c_2 *depending on* L *such that for any* $\epsilon > 0$ *and* $m \geq 1$ *satisfying*

$$m \geq c_1 \frac{s^2 n + s^2 \log p}{\epsilon^2}, \tag{6.136}$$

we have

$$\mathbb{E}\ \omega_2(A, s) \geq 1 - \epsilon, \tag{6.137}$$

$$\mathbb{P}\{\omega_2(A, s) \geq 1 - \epsilon\} \geq 1 - \exp(-c_2 \epsilon^2 m), \tag{6.138}$$

and

$$\mathbb{E}\ \omega_{\mathrm{b}\infty}(A^T A, s) \geq \frac{(1-\epsilon)^2}{s}, \tag{6.139}$$

$$\mathbb{P}\left\{\omega_{\mathrm{b}\infty}(A^T A, s) \geq \frac{(1-\epsilon)^2}{s}\right\} \geq 1 - \exp(-c_2 \epsilon^2 m). \tag{6.140}$$

Equation (6.136) and Theorem 6.1.1 imply that for exact signal recovery in the noise free case, we need $O(s^2(n + \log p)$ measurements for random sensing matrices. The extra s suggests that the $\omega_\diamond$ based approach to verify exact recovery is not as optimal as the one based on ρ_s. However, $\omega_\diamond$ is computational more amenable as we are going to see in Section 6.3. The measurement bound (6.136) also implies that the algorithms for verifying $\omega_\diamond > 0$ and for computing $\omega_\diamond$ work for s at least up to the order $\sqrt{m/(n + \log p)}$.

6.5 Preliminary Numerical Simulations

In this section, we present preliminary numerical results that assess the performance of the algorithms for verifying $\omega_2(A, s) > 0$ and computing $\omega_2(A, s)$. We also compare the error bounds based on $\omega_2(A, s)$ with the bounds based on the block RIP [6]. The involved semidefinite programs are solved using CVX.

We test the algorithms on Gaussian random matrices. The entries of Gaussian matrices are randomly generated from the standard Gaussian distribution. All $m \times np$ matrices are normalized to have columns of unit length.

We first present the values of s_* computed by (6.25), $k_* = \lfloor s_*/2 \rfloor$, and compare them with the corresponding quantities when A is seen as the sensing matrix for the sparse model without knowing the block-sparsity structure. The quantities in the later case are computed using the algorithms developed in [87, 93]. We note in Table 6.1 that for the same sensing matrix A, both s_* and k_* are smaller when the block-sparse structure is taken into account than it is not taken into account. However, we need to keep in mind that the true sparsity level in the block-sparse model is nk, where k is the block sparsity level. The nk_* in the fourth column for the block-sparse model is indeed much greater than the k_* in the sixth column for the sparse model, implying exploiting the block-sparsity structure is advantageous.

Table 6.1: Comparison of the sparsity level bounds on the block-sparse model and the sparse model for a Gaussian matrix $A \in \mathbb{R}^{m \times np}$ with $n = 4, p = 60$.

m	Block Sparse Model			Sparse Model	
	s_*	k_*	nk_*	s_*	k_*
72	3.96	1	4	6.12	3
96	4.87	2	8	7.55	3
120	5.94	2	8	9.54	4
144	7.14	3	12	11.96	5
168	8.60	4	16	14.66	7
192	11.02	5	20	18.41	9

In the next set of experiments, we compare the computation times for the three implementing methods discussed at the end of Section 6.3.5. The Gaussian matrix A is of size 72×120 with $n = 3$ and $p = 40$. The tolerance parameter is 10^{-5}. The initial η value for the Naive Fixed-Point Iteration is 0.1. The initial lower bound η_{L} and upper bound η_{U} are set as 0.1 and 10, respectively. All three implementations yields $\eta^* = 0.7034$. The CPU times for the three methods are 393 seconds, 1309 seconds, and 265 seconds. Therefore, the Fixed-Point Iteration + Bisection gives the most efficient implementation in general.

In the last experiment, we compare our recovery error bounds on the BS-BP based on $\omega_2(A, s)$ with those based on the block RIP. Recall the from Corollary 6.1.2, we have for the BS-BP

$$\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2 \leq \frac{2\sqrt{2k}}{\omega_2(A, 2k)}\varepsilon. \tag{6.141}$$

For comparison, the block RIP bounds is

$$\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2 \leq \frac{4\sqrt{1 + \delta_{2k}(A)}}{1 - (1 + \sqrt{2})\delta_{2k}(A)}\varepsilon, \tag{6.142}$$

assuming the block RIP $\delta_{2k}(A) < \sqrt{2} - 1$ [6]. Without loss of generality, we set $\varepsilon = 1$.

The block RIP is computed using Monte Carlo simulations. More explicitly, for $\delta_{2k}(A)$, we randomly take 1000 sub-matrices of $A \in \mathbb{R}^{m \times np}$ of size $m \times 2nk$ with a pattern determined by the block-sparsity structure, compute the maximal and minimal singular values σ_1 and σ_{2k}, and approximate $\delta_{2k}(A)$ using the maximum of $\max(\sigma_1^2 - 1, 1 - \sigma_{2k}^2)$ among all sampled sub-matrices. Obviously, the approximated block RIP is always smaller than or equal to the exact block RIP. As a consequence, the performance bounds based on the exact block RIP are *worse* than those based on the approximated block RIP. Therefore, in cases where our $\omega_2(A, 2k)$ based bounds are better (tighter, smaller) than the approximated block RIP bounds, they are even better than the exact block RIP bounds.

In Tables 6.2, we present the values of $\omega_2(A, 2k)$ and $\delta_{2k}(A)$ computed for a Gaussian matrix $A \in \mathbb{R}^{m \times np}$ with $n = 4$ and $p = 60$. The corresponding s_* and k_* for different m are also included in the table. We note that in all the considered cases, $\delta_{2k}(A) > \sqrt{2} - 1$, and the block RIP based bound (6.142) does not apply at all. In contrast, the ω_2 based bound (6.141) is valid as long as $k \leq k_*$. In Table (6.3) we show the ω_2 based bound 6.141.

Table 6.2: $\omega_2(A, 2k)$ and $\delta_{2k}(A)$ computed for a Gaussian matrix $A \in \mathbb{R}^{m \times np}$ with $n = 4$ and $p = 60$.

	m	72	96	120	144	168	192
	s_*	3.88	4.78	5.89	7.02	8.30	10.80
k	k_*	1	2	2	3	4	5
1	$\omega_2(A, 2k)$	0.45	0.53	0.57	0.62	0.65	0.67
	$\delta_{2k}(A)$	0.90	0.79	0.66	0.58	0.55	0.51
2	$\omega_2(A, 2k)$		0.13	0.25	0.33	0.39	0.43
	$\delta_{2k}(A)$		1.08	0.98	0.96	0.84	0.75
3	$\omega_2(A, 2k)$				0.11	0.18	0.25
	$\delta_{2k}(A)$				1.12	1.01	0.93
4	$\omega_2(A, 2k)$					0.02	0.12
	$\delta_{2k}(A)$					1.26	1.07
5	$\omega_2(A, 2k)$						0.03
	$\delta_{2k}(A)$						1.28

Table 6.3: The $\omega_2(A, 2k)$ based bounds on the ℓ_2 norms of the errors of the BS-BP for the Gaussian Matrix in Table 6.2.

	m	72	96	120	144	168	192
	s_*	3.88	4.78	5.89	7.02	8.30	10.80
k	k_*	1	2	2	3	4	5
1	ω_2 bound	6.22	13.01	9.89	6.50	11.52	9.50
2	ω_2 bound		58.56	25.37	14.64	7.30	16.26
3	ω_2 bound				53.54	21.63	30.27
4	ω_2 bound					236.74	23.25
5	ω_2 bound						127.59

Chapter 7

Low-Rank Matrix Recovery: Background

The last low-dimensional structure we will discuss in this dissertation is the low-rankness of matrices. We review previous work on low-rank matrix recovery in this chapter.

7.1 Introduction to Low-Rank Matrix Recovery

Suppose $X \in \mathbb{R}^{n_1 \times n_2}$ is a matrix of rank $r \ll \min\{n_1, n_2\}$, the low-rank matrix reconstruction problem aims to recover matrix X from a set of linear measurements $\boldsymbol{y}$ corrupted by noise $\boldsymbol{w}$:

$$\boldsymbol{y} \;\; = \;\; \mathcal{A}(X) + \boldsymbol{w}, \tag{7.1}$$

where $\mathcal{A} : \mathbb{R}^{n_1 \times n_2} \to \mathbb{R}^m$ is a linear measurement operator. Without loss of generality, we always assume $n_1 \leq n_2$. The noise vector $\boldsymbol{w} \in \mathbb{R}^m$ is either deterministic or random. In the deterministic setting we assume boundedness: $\|\boldsymbol{w}\|_2 \leq \varepsilon$, while in the stochastic setting we assume Gaussianity: $\boldsymbol{w} \sim \mathcal{N}(0, \sigma^2 \mathrm{I}_m)$. Since the matrix X lies in a low-dimensional sub-manifold of $\mathbb{R}^{n_1 \times n_2}$, we expect $m \ll n_1 n_2$ measurements would suffice to reconstruct X from $\boldsymbol{y}$ by exploiting the signal structure. Application areas of model (7.1) include factor analysis, linear system realization [94, 95], matrix completion [30, 31], quantum state tomography [96], face recognition [97, 98], Euclidean embedding [99], to name a few (See [7, 100, 101] for discussions and references therein).

A fundamental problem pertaining to model (7.1) is to reconstruct the low-rank matrix X from the measurement $\boldsymbol{y}$ by exploiting the low-rank property of X, and the stability of the reconstruction with respect to noise. For any reconstruction algorithm, we denote the estimate of X as $\hat{X}$, and the error matrix $H \overset{\text{def}}{=} \hat{X} - X$. The stability problem aims to bound $\|H\|_{\text{F}}$ in terms of m, n_1, n_2, r, the linear operator $\mathcal{A}$, and the noise strength ε or σ^2.

7.2 Recovery Algorithms

We briefly review three low-rank matrix recovery algorithms based on convex relaxation: the matrix Basis Pursuit, the matrix Dantzig selector, and the matrix LASSO estimator. A common theme of these algorithms is enforcing the low-rankness of solutions by penalizing large nuclear norms, or equivalently, the ℓ_1 norms of the singular value vectors. As a relaxation of the matrix rank, the nuclear norm remains a measure of low-rankness while being a convex function. In fact, the nuclear norm $\|\cdot\|_*$ is the convex envelop of rank$(\cdot)$ on the set $\{X \in \mathbb{R}^{n_1 \times n_2} : \|X\|_2 \leq 1\}$ [100, Theorem 2.2]. Most computational advantages of the aforementioned three algorithms result from the convexity of the nuclear norm.

The matrix Basis Pursuit algorithm [7, 100] tries to minimize the nuclear norm of solutions subject to the measurement constraint. It is applicable to both noiseless settings and bounded noise settings with a known noise bound ε. The matrix Basis Pursuit algorithm was originally developed for the noise-free case in [100]:

$$\text{mBP}: \quad \min_{Z \in \mathbb{R}^{n_1 \times n_2}} \|Z\|_* \ \text{ s.t. } \ \boldsymbol{y} = \mathcal{A}(Z), \tag{7.2}$$

which is a convex relaxation of the following rank minimization problem

$$\min_{Z \in \mathbb{R}^{n_1 \times n_2}} \text{rank}(Z) \ \text{ s.t. } \ \boldsymbol{y} = \mathcal{A}(Z). \tag{7.3}$$

The noisy version of matrix Basis Pursuit solves:

$$\text{mBP}: \quad \min_{Z \in \mathbb{R}^{n_1 \times n_2}} \|Z\|_* \ \text{ s.t. } \ \|\boldsymbol{y} - \mathcal{A}(Z)\|_2 \leq \epsilon. \tag{7.4}$$

In this dissertation, we refer to both cases as matrix Basis Pursuit.

The matrix Dantzig selector [7] reconstructs a low-rank matrix when its linear measurements are corrupted by unbounded noise. Its estimate for X is the solution to the nuclear norm regularization problem:

$$\text{mDS}: \quad \min_{Z \in \mathbb{R}^{n_1 \times n_2}} \|Z\|_* \ \text{ s.t. } \ \|\mathcal{A}^*(\boldsymbol{y} - \mathcal{A}^*(\boldsymbol{z}))\|_2 \leq \mu, \tag{7.5}$$

where μ a control parameter.

The matrix LASSO estimator solves the following optimization problem [7, 102]:

$$\text{mLASSO:} \quad \min_{Z \in \mathbb{R}^{n_1 \times n_2}} \frac{1}{2}\|\boldsymbol{y} - \mathcal{A}(Z)\|_2^2 + \mu\|Z\|_*. \tag{7.6}$$

All three optimization problems can be solved using convex programs. The key is to note that the nuclear norm admits an semidefinite programming characterization via

$$\|X\|_* = \min_{\substack{W^1 \in \mathbb{R}^{n_1 \times n_1} \\ W^2 \in \mathbb{R}^{n_2 \times n_2}}} \left(\text{trace}(W^1) + \text{trace}(W^2)\right)/2 \ \text{ s.t. } \ \begin{bmatrix} W^1 & X \\ X^T & W^2 \end{bmatrix} \succeq 0. \tag{7.7}$$

Recall from Chapter 2 that for any linear operator $\mathcal{A} : \mathbb{R}^{n_1 \times n_2} \mapsto \mathbb{R}^m$, its adjoint operator $\mathcal{A}^* : \mathbb{R}^m \mapsto \mathbb{R}^{n_1 \times n_2}$ is defined by the following relation

$$\langle \mathcal{A}(X), \boldsymbol{z} \rangle = \langle X, \mathcal{A}^*(\boldsymbol{z}) \rangle, \quad \forall X \in \mathbb{R}^{n_1 \times n_2}, \boldsymbol{z} \in \mathbb{R}^m. \tag{7.8}$$

The semidefinite representation of the nuclear norm (7.7) together with the following semidefinite representation for the constraint in the mDS (7.5)

$$\begin{bmatrix} \mu \mathrm{I}_{n_1} & \mathcal{A}^*(\boldsymbol{y} - \mathcal{A}(X)) \\ (\mathcal{A}^*(\boldsymbol{y} - \mathcal{A}(X)))^* & \mu \mathrm{I}_{n_2} \end{bmatrix} \succeq 0 \tag{7.9}$$

lead to an SDP formulation of the mDS:

$$
\begin{array}{cl}
\min\limits_{\substack{W^1 \in \mathbb{R}^{n_1 \times n_1} \\ W^2 \in \mathbb{R}^{n_2 \times n_2}}} & \left(\operatorname{trace}(W^1) + \operatorname{trace}(W^2)\right)/2 \\
\text{s.t.} & \begin{bmatrix} W^1 & Z & \mathbf{0} & \mathbf{0} \\ Z^T & W^2 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mu \mathrm{I}_{n_1} & \mathcal{A}^*(\boldsymbol{y} - \mathcal{A}(X)) \\ \mathbf{0} & \mathbf{0} & (\mathcal{A}^*(\boldsymbol{y} - \mathcal{A}(X)))^* & \mu \mathrm{I}_{n_2} \end{bmatrix} \succeq 0.
\end{array} \tag{7.10}
$$

The other two algorithms mBP and mLASSO can be recast into semidefinite programs in a similar manner.

7.3 Null Space Characterization and Restricted Isometry Property

The NSP for sparsity recovery and block-sparsity recovery has been extended to low-rank matrix recovery [38]. We first consider the rank minimization problem (7.3). Apparently, (7.3) correctly recovers X from $\boldsymbol{y} = \mathcal{A}(X)$ for any X with $\operatorname{rank}(X) = r$ if and only if

$$\operatorname{rank}(Z) \geq 2r, \forall\ Z \in \operatorname{null}(\mathcal{A}). \tag{7.11}$$

As shown in [38], the nuclear norm minimization problem (7.2) uniquely recovers X from $\boldsymbol{y} = \mathcal{A}(X)$ for any X with $\operatorname{rank}(X) = r$ if and only if

$$\|Z_1\|_* < \|Z_2\|_*, \forall\ Z = Z_1 + Z_2 \in \operatorname{null}(\mathcal{A}), \operatorname{rank}(Z_1) = r, \operatorname{rank}(Z_2) \geq r. \tag{7.12}$$

The aim of stability analysis is to derive error bounds of the solutions of the convex relaxation algorithms (7.4), (7.5), and (7.6). These bounds usually involve the incoherence of the linear operator $\mathcal{A}$, which is measured by the matrix restricted isometry constant (mRIC) defined below [7, 100]:

Definition 7.3.1. *For each integer $r \in \{1, \dots, n_1\}$, the matrix restricted isometry constant (mRIC) $\delta_{\mathrm{m}r}$ of a linear operator $\mathcal{A} : \mathbb{R}^{n_1 \times n_2} \mapsto \mathbb{R}^m$ is defined as the smallest $\delta > 0$ such that*

$$1 - \delta \leq \frac{\|\mathcal{A}(X)\|_2^2}{\|X\|_{\mathrm{F}}^2} \leq 1 + \delta \tag{7.13}$$

holds for arbitrary non-zero matrix X of rank at most r.

A linear operator $\mathcal{A}$ with a small δ_r means that $\mathcal{A}$ is nearly an isometry when restricted onto all matrices with rank at most r. Hence, it is no surprise that the mRIC is involved in the stability of recovering X from $\mathcal{A}(X)$ corrupted by noise when X is of rank at most r.

Now we cite stability results on the mBP, the mDS, and the mLASSO, which are expressed in terms of the mRIC. Assume X is of rank r and $\hat{X}$ is its estimate given by any of the three algorithms; then we have the following:

1. mBP [7]: *Suppose that $\delta_{4r} < \sqrt{2} - 1$ and $\|\boldsymbol{w}\|_2 \leq \varepsilon$. The solution to the mBP* (7.4) *satisfies*

$$\|\hat{X} - X\|_{\mathrm{F}} \leq \frac{4\sqrt{1 + \delta_{4r}}}{1 - (1 + \sqrt{2})\delta_{4r}} \cdot \varepsilon. \tag{7.14}$$

2. mDS [7]: *If $\delta_{4r} < \sqrt{2} - 1$ and $\|\mathcal{A}^*(\boldsymbol{w})\|_2 \leq \lambda$, then the solution to the mDS* (7.5)

$$\|\hat{X} - X\|_{\mathrm{F}} \leq \frac{16}{1 - (\sqrt{2} + 1)\delta_{4r}} \cdot \sqrt{r} \cdot \mu. \tag{7.15}$$

3. mLASSO [7]: *If $\delta_{4r} < (3\sqrt{2} - 1)/17$ and $\|\mathcal{A}^*(\boldsymbol{w})\| \leq \mu/2$, then the solution to the mLASSO* (7.6) *satisfies*

$$\|\hat{X} - X\|_{\mathrm{F}} \leq C(\delta_{4r})\sqrt{r} \cdot \mu, \tag{7.16}$$

for some numerical constant C.

7.4 Probabilistic Analysis

Although the mRIC provides a measure quantifying the goodness of a linear operator, its computation poses great challenges. In the literature, the computation issue is circumvented by resorting to a random argument. We cite one general result below [7]:

Let $\mathcal{A} : \mathbb{R}^{n_1 \times n_2} \mapsto \mathbb{R}^m$ be a random linear operator satisfying the concentration inequality for any $X \in \mathbb{R}^{n_1 \times n_2}$ and $0 < \epsilon < 1$:

$$\mathbb{P}\left(\left|\|\mathcal{A}(X)\|_2^2 - \|X\|_{\mathrm{F}}^2\right| \geq \epsilon \|X\|_{\mathrm{F}}^2\right) \leq Ce^{-mc_0(\epsilon)}. \tag{7.17}$$

for fixed constant $C > 0$. Then, for any given $\delta \in (0,1)$, there exist constants $c_1, c_2 > 0$ depending only on δ such that $\delta_r \leq \delta$, with probability not less than $1 - Ce^{-c_1 m}$, as long as

$$m \geq c_2 n r. \tag{7.18}$$

Chapter 8

Partial Extension to Low-Rank Matrix Recovery

We would like to extend the performance analysis and computational algorithms developed for sparsity recovery and block-sparsity recovery to low-rank matrix recovery. However, although we could develop a parallel theory for performance bounds and probabilistic analysis on the bounds, it turns out extremely difficult to extend the computational algorithms. In this chapter, we present the performance bounds based on the so called ℓ_* constrained minimal singular value and its probabilistic analysis. We leave the computational issues to future work. Without lass of generality, in this chapter we assume $n_1 \leq n_2$.

8.1 ℓ_*-Constrained Minimal Singular Values

We first introduce a quantity that continuously extends the concept of rank for a given matrix X. It is also an extension of the ℓ_1-sparsity level from vectors to matrices [103].

Definition 8.1.1. *The ℓ_*-rank of a non-zero matrix $X \in \mathbb{R}^{n_1 \times n_2}$ is defined as*

$$\tau(X) = \frac{\|X\|_*^2}{\|X\|_{\mathrm{F}}^2} = \frac{\|\boldsymbol{\sigma}(X)\|_1^2}{\|\boldsymbol{\sigma}(X)\|_2^2}, \tag{8.1}$$

where $\|\cdot\|_*$ is the nuclear norm of a matrix and $\|\cdot\|_{\mathrm{F}}$ the Frobenius norm. We use $\boldsymbol{\sigma}(X) \in \mathbb{R}^{n_1}$ to denote the vector of singular values of X in decreasing order.

The scaling invariant $\tau(X)$ is indeed a measure of rank. To see this, suppose $\mathrm{rank}(X) = r$; then Cauchy-Schwarz inequality implies that

$$\tau(X) \leq r, \tag{8.2}$$

and we have equality if and only if all non-zero singular values of X are equal. Therefore, the more non-zero singular values X has and the more evenly the magnitudes of these non-zero singular values are distributed, the larger $\tau(X)$. In particular, if X is of rank 1, then $\tau(X) = 1$; if X is of full rank n_1 with all singular values having the same magnitudes, then $\tau(X) = n_1$. However, if X has n_1 non-zero singular values but their magnitudes are spread in a wide range, then its ℓ_*-rank might be very small.

The ℓ_*-constrained minimal singular value is defined as follows:

Definition 8.1.2. *For any $\tau \in [1, n_1]$ and any linear operator $\mathcal{A} : \mathbb{R}^{n_1 \times n_2} \mapsto \mathbb{R}^m$, define the ℓ_*-constrained minimal singular value (abbreviated as ℓ_*-CMSV) by*

$$\rho_\tau(\mathcal{A}) \overset{\text{def}}{=} \inf_{X \neq 0,\ \tau(X) \leq \tau} \frac{\|\mathcal{A}(X)\|_2}{\|X\|_{\mathrm{F}}}. \tag{8.3}$$

For an operator $\mathcal{A}$, a non-zero $\rho_\tau(\mathcal{A})$ roughly means that $\mathcal{A}$ is invertible when restricted onto the set $\{X \in \mathbb{R}^{n_1 \times n_2} : \tau(X) \leq \tau\}$, or equivalently, the intersection of the null space of $\mathcal{A}$ and $\{X \in \mathbb{R}^{n_1 \times n_2} : \tau(X) \leq \tau\}$ contains only the null vector of $\mathbb{R}^{n_1 \times n_2}$. The value of $\rho_\tau(\mathcal{A})$ measures the invertibility of $\mathcal{A}$ restricted onto $\{\tau(X) \leq \tau\}$. As We will show that the error matrices for convex relaxation algorithms have small ℓ_*-ranks. Therefore, the error matrix is distinguishable from the zero matrix given the image of the error matrix under $\mathcal{A}$. Put it another way, given noise corrupted $\mathcal{A}(X)$, a signal matrix X is distinguishable from $X + H$, as long as the noise works in a way such that the error matrix H has a small ℓ_*-rank. This explains roughly why $\rho_\tau(\mathcal{A})$ determines the performance of convex relaxation algorithms.

8.1.1 Stability of Convex Relaxation Algorithms

In this section, we present the stability results for three convex relaxation algorithms: the matrix Basis Pursuit, the matrix Dantzig Selector, and the matrix LASSO estimator. As one will see in the proofs to Theorems 8.1.1, 8.1.2 and 8.1.3, the procedure of establishing these theorems has two steps:

1. Show that the error matrix $H = \hat{X} - X$ has a small ℓ_*-rank: $\tau(H) \leq \tau$ for some suitably selected τ, which automatically leads to a lower bound $\|\mathcal{A}(H)\|_2 \geq \rho_\tau \|H\|_{\mathrm{F}}$. Here X is the true matrix and $\hat{X}$ is its estimate given by convex relaxation algorithms.
2. Obtain an upper bound on $\|\mathcal{A}(H)\|_2$.

These are all relatively easy to show for the matrix Basis Pursuit algorithm. We have the following stability result:

Theorem 8.1.1. *If matrix X has rank r and the noise $\boldsymbol{w}$ is bounded; that is, $\|\boldsymbol{w}\|_2 \leq \epsilon$, then the solution $\hat{X}$ to the mBP* (7.4) *obeys*

$$\|\hat{X} - X\|_{\mathrm{F}} \leq \frac{2\epsilon}{\rho_{8r}}. \tag{8.4}$$

The corresponding bound (7.14) using mRIC is expressed as $\frac{4\sqrt{1+\delta_{4r}}}{1-(1+\sqrt{2})\delta_{4r}} \cdot \epsilon$ under the condition $\delta_{4r} \leq \sqrt{2} - 1$. Here δ_r is the mRIC defined in Definition 7.3.1. We note the ℓ_*-CMSV bound (8.4) is more concise and only requires $\rho_{8r} > 0$. Of course, we pay a price by replacing the subscript $4r$ with $8r$. A similar phenomena is also observed in the sparse signal reconstructions case. By analogy to the sparse case and the block-sparse case we expect that it is easier to get $\rho_{8r} > 0$ than $\delta_{4r} \leq \sqrt{2} - 1$.

Before stating the results for the matrix Dantzig Selector and the matrix Lasso estimator, we cite a lemma of [7]:

Lemma 8.1.1. [7, Lemma 1.1] *Suppose $\boldsymbol{w} \sim \mathcal{N}(0, \sigma^2 \mathrm{I}_m)$. If $C \geq 4\sqrt{(1 + \delta_{\mathrm{m1}}\max(\mathcal{A})) \log 12}$, then there exists a numerical constant $c > 0$ such that with probability greater than*

$1 - 2\exp(-cn_2)$ that

$$\|\mathcal{A}^*(\boldsymbol{w})\| \leq C\sqrt{n_2}\sigma, \tag{8.5}$$

where $\mathcal{A}^*$ is the adjoint operator of $\mathcal{A}$.

Lemma 8.1.1 allows to transform statements under the condition of $\|\mathcal{A}^*(\boldsymbol{w})\|_2 \leq \mu$, *e.g.* Theorem 8.1.2 and 8.1.3, into ones that hold with large probability. We now present the error bounds for the matrix Dantzig Selector and the matrix LASSO estimator, whose proofs can be found in Sections 8.1.3 and 8.1.4, respectively.

Theorem 8.1.2. *Suppose the noise vector in model* (7.1) *satisfies $\|\mathcal{A}^*(\boldsymbol{w})\|_2 \leq \mu$, and suppose $X \in \mathbb{R}^{n_1 \times n_2}$ is of rank r. Then, the solution $\hat{X}$ to the mDS* (7.5) *satisfies*

$$\|\hat{X} - X\|_{\mathrm{F}} \leq \frac{4\sqrt{2}}{\rho_{8r}^2} \cdot \sqrt{r} \cdot \mu. \tag{8.6}$$

Theorem 8.1.3. *Suppose the noise vector in model* (7.1) *satisfies $\|\mathcal{A}^*(\boldsymbol{w})\|_2 \leq \kappa\mu$ for some $\kappa \in (0, 1)$, and suppose $X \in \mathbb{R}^{n_1 \times n_2}$ is of rank r. Then, the solution $\hat{X}$ to the matrix LASSO estimator* (7.6) *satisfies*

$$\|\hat{X} - X\|_{\mathrm{F}} \leq \frac{1+\kappa}{1-\kappa} \cdot \frac{2\sqrt{2}}{\rho_{\frac{8r}{(1-\kappa)^2}}^2} \cdot \sqrt{r} \cdot \mu. \tag{8.7}$$

For example, if we take $\kappa = 1 - 2\sqrt{2}/3$, then the bound becomes

$$\|\hat{X} - X\|_{\mathrm{F}} \leq 6(1 - \frac{\sqrt{2}}{3}) \cdot \frac{1}{\rho_{9r}^2} \cdot \sqrt{r} \cdot \mu. \tag{8.8}$$

The readers are encouraged to compare the statements of Theorem 8.1.2 and 8.1.3 with those using mRIC as cited in Section 7.3 (Equations (7.15), (7.16) and the conditions for them to be valid).

In this section, we present the derivation of bounds on the reconstruction error for the matrix Basis Pursuit, the matrix Dantzig selector and the matrix LASSO estimator. As shown in Theorems 8.1.1, 8.1.2 and 8.1.3, our bounds are given in terms of the ℓ_*-CMSV rather than the mRIC of linear operator $\mathcal{A}$.

8.1.2 Basis Pursuit

We first establish a bound on the Frobenius norm of mBP error matrix using the ℓ_*-CMSV. Recall the two steps discussed in Section 8.1.1:

1. Show that the error matrix $H = \hat{X} - X$ has small ℓ_* rank: $\tau(H) \leq 8r$, which automatically leads to a lower bound $\|\mathcal{A}(H)\|_2 \geq \rho_{8r}\|H\|_{\mathrm{F}}$;
2. Obtain an upper bound on $\|\mathcal{A}(H)\|_2$.

For mBP (7.4), the second step is trivial as both X and $\hat{X}$ satisfy constraint $\|\boldsymbol{y} - \mathcal{A}(Z)\| \leq \epsilon$ in (7.4). Therefore, the triangle inequality yields

$$\begin{aligned} \|\mathcal{A}(H)\|_2 &= \|\mathcal{A}(\hat{X} - X)\|_2 \\ &\leq \|\mathcal{A}(\hat{X}) - \boldsymbol{y}\|_2 + \|\boldsymbol{y} - \mathcal{A}(X)\|_2 \\ &\leq 2\epsilon. \end{aligned} \tag{8.9}$$

In order to establish that the error matrix has a small ℓ_*-rank in the first step, we present two lemmas on the properties of nuclear norms derived in [100]:

Lemma 8.1.2. [100, Lemma 2.3] *Let A and B be matrices of the same dimensions. If $AB^T = 0$ and $A^TB = 0$ then $\|A + B\|_* = \|A\|_* + \|B\|_*$.*

Lemma 8.1.3. [100, Lemma 3.4] *Let A and B be matrices of the same dimensions. Then there exist matrices B^1 and B^2 such that*

1. $B = B^1 + B^2$
2. $\mathrm{rank}(B^1) \leq 2\mathrm{rank}(A)$
3. $AB^{2T} = 0$ *and* $A^TB^2 = 0$
4. $\langle B^1, B^2\rangle = 0$.

Now we give a proof of Theorem 8.1.1:

Proof of Theorem 8.1.1. We decompose the error matrix $B = H$ according to Lemma 8.1.3 with $A = X$, more explicitly, we have:

1. $H = H^0 + H^c$
2. $\text{rank}(H^0) \leq 2\text{rank}(X) = 2r$
3. $XH^{cT} = 0$ and $X^T H^c = 0$
4. $\langle H^0, H^c \rangle = 0$.

As observed by Recht *et.al* in [100] (See also [39], [7] and [103]), the fact that $\|\hat{X}\|_* = \|X + H\|_1$ is the minimum among all Zs satisfying the constraint in (7.4) implies that $\|H^c\|_*$ cannot be very large. To see this, we observe that

$$\begin{aligned} \|X\|_* &\geq \|X + H\|_* \\ &= \|X + H^c + H^0\|_* \\ &\geq \|X + H^c\|_* - \|H^0\|_* \\ &= \|X\|_* + \|H^c\|_* - \|H^0\|_*. \end{aligned} \tag{8.10}$$

Here, for the last equality we used Lemma 8.1.2 and $XH^{cT} = 0, X^T H^c = 0$. Therefore, we obtain

$$\|H^c\|_* \leq \|H^0\|_*, \tag{8.11}$$

which leads to

$$\begin{aligned} \|H\|_* &\leq \|H^0\|_* + \|H^c\|_* \\ &\leq 2\|H^0\|_* \\ &\leq 2\sqrt{\text{rank}(H^0)}\|H^0\|_{\text{F}} \\ &= 2\sqrt{2r}\|H\|_{\text{F}}, \end{aligned} \tag{8.12}$$

where for the next to the last inequality we used the fact that $\|H\|_* \leq \sqrt{\text{rank}(H)}\|H\|_F$, and for the last inequality we used the pythagoras theorem $\|H\|_{\text{F}}^2 = \|H^0\|_{\text{F}}^2 + \|H^c\|_{\text{F}}^2 \geq$

$\|H^0\|_{\mathrm{F}}^2$ because $\langle H^0, H^c \rangle = 0$. Inequality (8.12) is equivalent to

$$\tau(H) \leq 8 \operatorname{rank}(X) = 8r. \tag{8.13}$$

It follows from (8.9) and Definition 8.1.2 that

$$\rho_{8r}\|H\|_{\mathrm{F}} \;\; \leq \;\; \|\mathcal{A}(H)\|_2 \leq 2\epsilon. \tag{8.14}$$

Hence, we get the conclusion of Theorem 8.1.1

$$\|\hat{X} - X\|_{\mathrm{F}} \;\; \leq \;\; \frac{2\epsilon}{\rho_{8r}}. \tag{8.15}$$

□

8.1.3 Dantzig Selector

This subsection is devoted to the proof of Theorem 8.1.2.

Proof of Theorem 8.1.2. Suppose $X \in \mathbb{R}^{n_1 \times n_2}$ is of rank r, and $\hat{X}$ is the solution to the matrix Dantzig selector (7.5). Define $H = \hat{X} - X$. We note that to obtain that H has a small ℓ_*-rank (8.13), we used only two conditions:

- $\|\hat{X}\|_* = \|X + H\|_*$ is the minimum among all matrices satisfying the optimization constraint;
- the true signal X satisfies the constraint.

Obviously, the first condition holds simply because of the structure of the matrix Dantzig selector. If the noise vector $\boldsymbol{w}$ satisfies $\|\mathcal{A}(\boldsymbol{w})\|_2 \leq \mu$, then the true signal X also satisfy the constraint:

$$\begin{aligned} \|\mathcal{A}^*(\boldsymbol{r})\|_2 &= \|\mathcal{A}^*(\boldsymbol{y} - \mathcal{A}(X))\|_2 \\ &= \|\mathcal{A}^*(\boldsymbol{w})\|_2 \leq \mu. \end{aligned} \tag{8.16}$$

Consequently, we have $\tau(H) \leq 8r$ following the same procedure as in the Proof of Theorem 8.1.1 in Section 8.1.2, or equivalently,

$$\|H\|_* \leq \sqrt{8r}\|H\|_{\mathrm{F}}. \tag{8.17}$$

We now turn to the second step to obtain an upper bound on $\|\mathcal{A}(X)\|_{\mathrm{F}}$. The condition $\|\mathcal{A}(\boldsymbol{w})\|_2 \leq \mu$ and the constraint in the Dantzig selector (7.5) yield

$$\|\mathcal{A}^*(\mathcal{A}(H))\|_2 \ \ \leq \ \ 2\mu \tag{8.18}$$

because

$$\begin{aligned} \mathcal{A}^*(\boldsymbol{w} - \hat{\boldsymbol{r}}) = \mathcal{A}^* \Big((\boldsymbol{y} - \mathcal{A}(X)) - (\boldsymbol{y} - \mathcal{A}(\hat{X}))\Big) \\ = \ \mathcal{A}^* \Big(\mathcal{A}(\hat{X}) - \mathcal{A}(X)\Big) = \mathcal{A}^*(\mathcal{A}(H)), \end{aligned} \tag{8.19}$$

where $\hat{\boldsymbol{r}} = \boldsymbol{y} - \mathcal{A}(\hat{X})$ is the residual corresponding to the matrix Dantzig selector solution $\hat{X}$. Therefore, we obtain an upper bound on $\|\mathcal{A}(H)\|_{\mathrm{F}}^2$ as follows:

$$\begin{aligned} \langle \mathcal{A}(H), \mathcal{A}(H)\rangle &= \langle H, \mathcal{A}^*(\mathcal{A}(H))\rangle \\ &\leq \|H\|_*\|\mathcal{A}^*(\mathcal{A}(H))\|_2 \\ &\leq 2\mu\|H\|_*. \end{aligned} \tag{8.20}$$

Equation (8.20), the definition of ρ_{8r}, and equation (8.17) together yield

$$\begin{aligned} \rho_{8r}^2\|H\|_{\mathrm{F}}^2 &\leq \langle \mathcal{A}(H), \mathcal{A}(H)\rangle \\ &\leq 2\mu\|H\|_* \\ &\leq 2\mu\sqrt{8r}\sigma\|H\|_{\mathrm{F}}. \end{aligned} \tag{8.21}$$

We conclude that

$$\|H\|_{\mathrm{F}} \ \ \leq \ \ \frac{4\sqrt{2}}{\rho_{8r}^2} \cdot \sqrt{r} \cdot \mu, \tag{8.22}$$

which is exactly the result of Theorem 8.1.2. □

8.1.4 LASSO Estimator

We derive a bound on the matrix LASSO estimator using the procedure developed in [7] (see also [80]).

Proof of Theorem 8.1.3. Suppose the noise $\boldsymbol{w}$ satisfies $\|\mathcal{A}^*(\boldsymbol{w})\|_2 \leq \kappa\mu$ for some small $\kappa > 0$. Because $\hat{X}$ is a solution to (7.6), we have

$$\frac{1}{2}\|\mathcal{A}(\hat{X}) - \boldsymbol{y}\|_2^2 + \mu\|\hat{X}\|_* \leq \frac{1}{2}\|\mathcal{A}(X) - \boldsymbol{y}\|_2^2 + \mu\|X\|_*.$$

Consequently, substituting $\boldsymbol{y} = \mathcal{A}(X) + \boldsymbol{w}$ yields

$$\begin{aligned}
\mu\|\hat{X}\|_* &\leq \frac{1}{2}\|\mathcal{A}(X) - \boldsymbol{y}\|_2^2 - \frac{1}{2}\|\mathcal{A}(\hat{X}) - \boldsymbol{y}\|_2^2 + \mu\|X\|_* \\
&= \frac{1}{2}\|\boldsymbol{w}\|_2^2 - \frac{1}{2}\|\mathcal{A}(\hat{X} - X) - \boldsymbol{w}\|_2^2 + \mu\|X\|_* \\
&= \frac{1}{2}\|\boldsymbol{w}\|_2^2 - \frac{1}{2}\|\mathcal{A}(\hat{X} - X)\|_2^2 \\
&\qquad\qquad + \left\langle \mathcal{A}(\hat{X} - X), \boldsymbol{w} \right\rangle - \frac{1}{2}\|\boldsymbol{w}\|_2^2 + \mu\|X\|_* \\
&\leq \left\langle \mathcal{A}(\hat{X} - X), \boldsymbol{w} \right\rangle + \mu\|X\|_* \\
&= \left\langle \hat{X} - X, \mathcal{A}^*(\boldsymbol{w}) \right\rangle + \mu\|X\|_*.
\end{aligned}$$

Using the Cauchy-Swcharz type inequality, we get

$$\begin{aligned}
\mu\|\hat{X}\|_* &\leq \|\hat{X} - X\|_*\|\mathcal{A}^*(\boldsymbol{w})\|_2 + \mu\|X\|_* \\
&= \kappa\mu\|H\|_* + \mu\|X\|_*,
\end{aligned}$$

which leads to

$$\|\hat{X}\|_* \leq \kappa\|H\|_* + \|X\|_*.$$

Therefore, similar to the argument in (8.10) we have

$$\begin{aligned}\|X\|_* &\geq \|\hat{X}\|_* - \kappa\|H\|_* \\ &= \|X+H\|_* - \kappa\|H\|_* \\ &\geq \|X+H^c+H^0\|_* - \kappa\left(\|H^c\|_* + \|H^0\|_*\right) \\ &\geq \|X+H^c\|_* - \|H^0\|_* - \kappa\left(\|H^c\|_* + \|H^0\|_*\right) \\ &= \|X\|_* + \|H^c\|_* - \|H^0\|_* - \kappa\left(\|H^c\|_* + \|H^0\|_*\right) \\ &= \|X\|_* + (1-\kappa)\|H^c\|_* - (1+\kappa)\|H^0\|_*.\end{aligned}$$

Consequently, we have

$$\|H^c\|_* \leq \frac{1+\kappa}{1-\kappa}\|H^0\|_*,$$

an inequality slightly worse than (8.11) for small κ. Therefore, an argument similar to the one leading to (8.12) yields

$$\|H\|_* \leq \frac{2}{1-\kappa}\sqrt{2r}\|H\|_{\mathrm{F}}, \tag{8.23}$$

or equivalently,

$$\tau(H) \leq \frac{8r}{(1-\kappa)^2}. \tag{8.24}$$

Now we need to establish a bound on

$$\begin{aligned}\|\mathcal{A}^*(\mathcal{A}(H))\|_2 &\leq \|\mathcal{A}^*(\boldsymbol{y}-\mathcal{A}(X))\|_2 + \|\mathcal{A}^*(\boldsymbol{y}-\mathcal{A}(\hat{X}))\|_2 \\ &\leq \|\mathcal{A}^*(\boldsymbol{w})\|_2 + \|\mathcal{A}^*(\boldsymbol{y}-\mathcal{A}(\hat{X}))\|_2 \\ &= \kappa\mu + \|\mathcal{A}^*(\boldsymbol{y}-\mathcal{A}(\hat{X}))\|_2. \end{aligned}\tag{8.25}$$

We follow the procedure in [7] (see also [80]) to estimate $\|\mathcal{A}^*(\boldsymbol{y}-\mathcal{A}(\hat{X}))\|_2$. Since $\hat{X}$ is the solution to (7.6), the optimality condition yields that

$$\mathcal{A}^*(\boldsymbol{y}-\mathcal{A}(\hat{X})) \in \partial\|\hat{X}\|_*, \tag{8.26}$$

where $\partial\|\hat{X}\|_*$ is the family of subgradient of $\|\cdot\|_*$ evaluated at $\hat{X}$. According to [31], if the singular value decomposition of $\hat{X}$ is $U\Sigma V^T$, then we have

$$\begin{aligned}\partial\|\hat{X}\|_* = \{\mu(UV^T + W) : \|W\|_2 \leq 1,\\ U^TW = 0, WV = 0\}.\end{aligned} \tag{8.27}$$

As a consequence, we obtain $\mathcal{A}^*(\boldsymbol{y} - \mathcal{A}(\hat{X})) = \mu(UV^T + W)$ and

$$\begin{aligned}\|\mathcal{A}^*(\boldsymbol{y} - \mathcal{A}(\hat{X}))\|_2 &\leq \|\mu(UV^T + W)\|_2\\ &= \mu.\end{aligned} \tag{8.28}$$

We used $\|UV^T + W\|_2 = 1$ because

$$\begin{aligned}&\max_{\boldsymbol{x}:\|\boldsymbol{x}\|_2=1} \|(UV^T + W)\boldsymbol{x}\|_2\\ =& \max_{\boldsymbol{y}:\|\boldsymbol{y}\|_2=1} \|(UV^T + W)V\boldsymbol{y}\|_2 \leq 1.\end{aligned} \tag{8.29}$$

Following the same lines in (8.20), we get

$$\|\mathcal{A}(H)\|_2^2 \leq (\kappa + 1)\mu\|H\|_*. \tag{8.30}$$

Then, Equation (8.23), (8.25) and (8.28)

$$\begin{aligned}&\rho^2_{\frac{8r}{(1-\kappa)^2}}\|H\|_\mathrm{F}^2 \leq \|\mathcal{A}(H)\|_2^2\\ \leq& (\kappa + 1)\mu\frac{\sqrt{8r}}{1-\kappa}\|H\|_\mathrm{F}.\end{aligned} \tag{8.31}$$

As a consequence, the conclusion of Theorem 8.1.3 holds. □

8.2 Probabilistic Analysis

This section is devoted to analyzing the properties of the ℓ_*-CMSVs for several important random sensing ensembles. Although the bounds in Theorem 8.1.1, 8.1.2 and 8.1.3 have concise forms, they are useless if the quantity involved, ρ_τ, is zero or approaches zero for most matrices as n_1, n_2, m, k vary in a reasonable manner. We show that, at least for the isotropic and subgaussian ensemble, the ℓ_*-CMSVs are bounded away from zero with high probability.

Recall from Chapter 2 that a linear operator $\mathcal{A} : \mathbb{R}^{n_1 \times n_2} \to \mathbb{R}^m$ can be represented by a collection of matrices $\mathscr{A} = \{A^1, \ldots, A^m\}$. Based on this representation of $\mathcal{A}$, we have the following definition of isotropic and subgaussian operators:

Definition 8.2.1. *Suppose $\mathcal{A} : \mathbb{R}^{n_1 \times n_2} \to \mathbb{R}^m$ is a linear operator with corresponding matrix representation $\mathscr{A}$. We say $\mathcal{A}$ is from the isotropic and subgaussian ensemble if for each $A^i \in \mathscr{A}$, $\text{vec}(A^i)$ is an independent isotropic and subgaussian vector with constant L, and L is a numerical constant independent of n_1, n_2.*

For any isotropic and subgaussian operator $\sqrt{m}\mathcal{A}$ the typical value of $\rho_\tau(\mathcal{A})$ concentrates around 1 for relatively large m (but $\ll n_1 n_2$). More precisely, we have the following theorem:

Theorem 8.2.1. *Let $\sqrt{m}\mathcal{A}$ be an isotropic and subgaussian operator with some numerical constant L. Then there exists absolute constants c_1, c_2 depending on L only such that for any $\epsilon > 0$ and $m \geq 1$ satisfying*

$$m \geq c_1 \frac{\tau n_2}{\epsilon^2}, \tag{8.32}$$

we have

$$\mathbb{E}|1 - \rho_\tau(\mathcal{A})| \leq \epsilon \tag{8.33}$$

and

$$\mathbb{P}\{1 - \epsilon \leq \rho_\tau(\mathcal{A}) \leq 1 + \epsilon\} \geq 1 - \exp(-c_2 \epsilon^2 m). \tag{8.34}$$

Proof of Theorem 8.2.1. Since the linear operator $\mathcal{A}$ is generated in a way such that $\mathbb{E}\|\mathcal{A}(X)\|_2^2 = \|X\|_{\text{F}}^2$ for any $X \in \mathbb{R}^{n_1 \times n_2}$, we have $|\rho_\tau(\mathcal{A}) - 1| < 1 - \epsilon$ is a consequence of

$$\begin{aligned} & \sup_{X\in\mathcal{H}_\tau}\left|\frac{1}{m}\mathcal{A}(X)^T\mathcal{A}(X)-1\right| \\ = & \sup_{X\in\mathcal{H}_\tau}\left|\frac{1}{m}\sum_{k=1}^{m}\left\langle A^k, X\right\rangle^2-1\right| \leq \epsilon. \end{aligned} \tag{8.35}$$

As usual, the operator $\mathcal{A}$ is represented by a collection of matrices $\mathscr{A} = \{A^1, \ldots, A^m\}$. We define a class of functions parameterized by X as $\mathcal{F}_\tau \overset{\text{def}}{=} \{f_X(\cdot) = \langle X, \cdot\rangle : X \in \mathcal{H}_\tau\}$ with

$$\mathcal{H}_\tau = \{X \in \mathbb{R}^{n_1\times n_2} : \|X\|_{\text{F}} = 1,\ \|X\|_*^2 \leq \tau\}. \tag{8.36}$$

It remains to compute $\ell_*(\mathcal{H}_\tau)$ as follows

$$\begin{aligned} \ell_*(\mathcal{H}_\tau) &= \mathbb{E}\sup_{X\in\mathcal{H}_\tau}\langle G, X\rangle \\ &\leq c\,\|X\|_*\,\mathbb{E}\,\|G\|_2 \\ &\leq c\,\sqrt{\tau}\sqrt{n_2}, \end{aligned} \tag{8.37}$$

where G is a Gaussian matrix with *i.i.d.* entries from $\mathcal{N}(0,1)$. As a consequence, the conclusions of Theorem 8.2.1 hold. □

8.3 Computational Difficulties

There are significant challenges in extending the computational framework developed for sparsity recovery and block-sparsity recovery to low-rank matrix recovery. One might attempt to use the spectral norm $\|X\|_2$ to define a similar ω function and develop a fixed point theory to compute it. The fixed point theory can indeed be generalized to low-rank matrix recovery. However, the subprograms to compute the associated scalar auxiliary functions are difficult to solve. To see this, note that the verification problems induced by the ℓ_*-CMSV and the ω function defined via the spectral norm take the form:

$$\max_{X\in\mathbb{R}^{n_1\times n_2}} \|X\|_\diamond \ \text{ s.t. } \ \mathcal{A}(X)=0, \|X\|_*\le 1, \tag{8.38}$$

where $\diamond = \mathrm{F}$ or 2. When $\diamond = \mathrm{F}$ and $X \in \mathbb{R}^{n\times n}$ is a diagonal matrix with $\boldsymbol{x}\in\mathbb{R}^n$ on its diagonal, (8.38) is equivalent with

$$\max_{\boldsymbol{x}\in\mathbb{R}^n} \|\boldsymbol{x}\|_2 \ \text{ s.t. } \ A\boldsymbol{x}=0, \|\boldsymbol{x}\|_1\le 1, \tag{8.39}$$

for suitably defined A. As we mentioned in Chapter 4, the optimization (8.39) is NP hard [84].

When $\diamond = 2$, it is also not obvious how to solve (8.38). A relaxation procedure similar to those for sparsity recovery and block-sparsity recovery goes as follows:

$$\begin{aligned}
& \max\{\|X\|_2 : X\in\mathbb{R}^{n_1\times n_2}, \mathcal{A}(X)=0, \|X\|_*\le 1\} \\
= \ & \max\{\langle X, Y\rangle : X\in\mathbb{R}^{n_1\times n_2}, Y\in\mathbb{R}^{n_1\times n_2}, \mathcal{A}(X)=0, \|X\|_*\le 1, \|Y\|_*\le 1\} \\
= \ & \max\{\left\langle X-\mathcal{B}^T\circ\mathcal{A}(X), Y\right\rangle : X\in\mathbb{R}^{n_1\times n_2}, Y\in\mathbb{R}^{n_1\times n_2}, \mathcal{A}(X)=0, \|X\|_*\le 1, \|Y\|_*\le 1\} \\
\le \ & \max\{\left\langle X-\mathcal{B}^T\circ\mathcal{A}(X), Y\right\rangle : X\in\mathbb{R}^{n_1\times n_2}, Y\in\mathbb{R}^{n_1\times n_2}, \|X\|_*\le 1, \|Y\|_*\le 1\} \\
= \ & \max \|(\mathrm{id}-\mathcal{B}^T\circ\mathcal{A})(X)\|_2 \ \text{ s.t. } \ \|X\|_*\le 1 \\
\overset{\text{def}}{=} \ & g(\mathcal{B}).
\end{aligned} \tag{8.40}$$

We have used $\circ$ to denote the composition of two operators and id to denote the identity operator. Then a tight upper bound is obtained by minimizing the following *convex* function

$$g(\mathcal{B}) = \max_{X\neq 0} \frac{\|(\mathrm{id}-\mathcal{B}^T\circ\mathcal{A})(X)\|_2}{\|X\|_*}, \tag{8.41}$$

which is the operator norm of $\mathrm{id}-\mathcal{B}^T\circ\mathcal{A} : (\mathbb{R}^{n_1\times n_2}, \|\cdot\|_*) \to (\mathbb{R}^{n_1\times n_2}, \|\cdot\|_2)$. Unfortunately, we do not have an efficient way to compute this operator norm and/or its subgradient.

Chapter 9

Conclusions and Future Work

9.1 Conclusions

In this dissertation, we analyzed the performance on recovering signals with low-dimensional structures, in particular, the sparsity, block-sparsity, and low-rankness. According to the addressed problems and the used techniques, the body of the work can be divided into three parts:

- **Goodness Measures and Performance Bounds:** In this part, we defined functions such as $\rho_s(A)$ and $\omega_\diamond(Q, s)$ to quantify the goodness of the sensing matrix in the context of signal recovery. These goodness measures are usually defined as the Rayleigh quotient type quantities minimized over a set characterizing the recovery error vectors. Therefore, the functions $\rho_s(A)$ and $\omega_\diamond(Q, s)$ quantify the invertibility of the sensing matrix A in the worst-case scenario. We then derived performance bounds in terms of these goodness measures. One distinct feature of this work is that, instead of the popular ℓ_2 norm, we used the ℓ_∞ norm and the block-ℓ_∞ norm as the performance criteria for sparsity recovery and block-sparsity recovery, respectively. We chose these two unconventional norms because of their connections with other performance criteria, especially the signal support, and because of the computability of the resulting goodness measures and performance bounds.
- **Computational Algorithms:** We designed algorithms to compute the goodness measures, more explicitly, the ω functions. We overcame the computational difficulty due to the highly non-convexity of ω by reducing its computation and

approximation to the localization of the unique positive fixed points of scalar auxiliary functions. These auxiliary functions were in turn computed by a series of linear programs, second-order cone programs, or semidefinite programs. Based on the fixed point characterization of the goodness measures, we developed efficient algorithms to compute the ω functions using fixed point iterations and bisection search, and established their convergence.

As a by-product, we developed algorithms to verify the uniqueness and exactness of various convex relaxations algorithms in the noise-free case. These verification algorithms are generally easier to solve than to compute or approximate the ω functions directly. Numerical simulations show that our implementations produce state-of-the-art results.

- **Probabilistic Analysis:** The third part of this work consists of probabilistic analysis of the goodness measures for random sensing matrices, especially the isotropic and subgaussian ensemble. We employed a general result in estimating the convergence rates of empirical processes, which was established using the idea of generic chaining. By reducing the problem to merely computing the expected value of the supreme of a properly defined Gaussian process, the empirical process approach turned out to be a truly powerful tool in analyzing the probabilistic behavior of the ω and ρ functions. As a matter of fact, these functions defined for sparsity recovery, block-sparsity recovery, and low-rank matrix recovery can be analyzed using the empirical process approach in a very similar manner.

We expect that these goodness measures will be useful in comparing different sensing systems and recovery algorithms, as well as in designing optimal sensing matrices.

9.2 Future Work

In future, we plan to extend the work in several directions expounded in the following.

- **Optimal Sensing Matrix Design:** We plan to apply our computational algorithms to the optimal design of the sensing matrix. A complete optimal design framework would require optimizing, *e.g.*, the ω functions subject to constraint on the allowable sensing matrices. Since the procedures to compute the ω functions are already complex, the problems of optimal design pose significant challenges. We will instead consider simpler problems. For example, we plan to consider how to extract a submatrix B from a large sensing matrix A with as fewer rows as possible but retain the recovery performance of A. Note that for block-sparsity recovery, the recovery performance is determined by

$$\begin{aligned} & \max_i \min_{P_{[i]}} s\eta \left(\max_j \|\delta_{ij} \mathrm{I}_n - P_{[i]}^T A_{[j]}\|_2 \right) + \|P_{[i]}\|_2, \\ = \quad & \|\mathrm{I}_{np} - P^{*T} A\|_{\mathrm{b}\ 2,\infty} + \max_i \|P_{[i]}^*\|_2, \end{aligned} \tag{9.1}$$

where P^* is the optimal solutions and the $\|\cdot\|_{\mathrm{b}\ 2,\infty}$ norm for an $np \times np$ matrix is defined as the maximal largest singular values of its p^2 $n \times n$ blocks. If we could select rows of A indexed by S to form a submatrix B such that

$$\|P^{*T} A - P_{[S]}^{*T} B\|_{\mathrm{b}\ 2,\infty} \tag{9.2}$$

is small, then triangle inequality implies that B might produce recovery performance similar to A. We therefore convert the problem of sensing matrix selection into one of low rank matrix approximation with the $\|\cdot\|_{\mathrm{b}\ 2,\infty}$ norm as the approximation accuracy measure. A similar idea was developed in [104] for sparsity recovery.

- **Extension to Other Low-dimensional Structures:** One common feature of the low-dimensional signals considered in this paper is that they can be expressed as the linear combinations of a few elements from an atomic set. For example, the atomic set for sparse signals is the set of the canonical basis vectors, and the atomic set for low-rank matrices is the set of all rank one matrices. This idea has been extended into other atomic sets such as those formed by sign-vector $\{-1, +1\}^n$, permutation matrices, and orthogonal matrices, and finds applications in many practical problems [105]. It would be interesting to extend the computable performance analysis to these new low-dimensional structures.

- **Extension to Nonuniform Recovery:** This work focuses on the performance analysis of uniform recovery. We would like to extend the results to nonuniform recovery. For uniform recovery, once the sensing matrix is chosen, then *all* signals with certain low-dimensional structure must be able to be recovered as long as the noise satisfies some conditions. Therefore, the performance analysis on uniform recovery is a worst-case-scenario analysis and the conditions imposed on the sensing matrix are quiet stringent. On the other hand, nonuniform recovery only requires the successful recovery of a portion of the signals, sometimes a single signal. For example, in sparsity recovery we might have some prior information of the support of the underlying signal, and require stable recovery only those signals described by the prior information. The prior information might come from previous estimates of the signals in a system where signals are streaming in and changing slowly. We expect that by properly incorporating the prior information we would obtain tighter performance bounds that are still computable.

- **Applications to practical problems:** We will apply our computable performance measures to many practical problems. Areas of interest include sensor array, MRI, MIMO radar, reflectance estimation, and machine learning. For example, in reflectance estimation from RGB information using sparsity recovery [106], the system performance can be greatly improved if we measure the RGB under several different illuminations. Since the choices of illuminations are very limited, we could compare their performance by directly computing the goodness measures associated with each choice. We expect the computable performance measures we developed in this dissertation would provide new approaches to many such practical problems.

References

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

[2] H. Chang, D. Y. Yeung, and Y. Xiong, “Super-resolution through neighbor embedding,” in *Proc. IEEE Conf. Comput. Vis*, 2004, vol. 1, pp. 275–282.

[3] J. Yang, J. Wright, T. S. Huang, and Y. Ma, “Image super-resolution via sparse representation,” *IEEE Trans. Image Process.*, vol. 19, no. 11, pp. 2861 –2873, nov. 2010.

[4] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma, “Robust face recognition via sparse representation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, feb 2009.

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

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

[7] E. J. Candès and Y. Plan, “Tight oracle inequalities for low-rank matrix recovery from a minimal number of noisy random measurements,” *IEEE Trans. Inf. Theory*, vol. 57, no. 4, pp. 2342–2359, apr 2011.

[8] J. B. Tenenbaum, V. de Silva, and J. C. Langford, “A global geometric framework for nonlinear dimensionality reduction,” *Science*, vol. 290, no. 5500, dec 2000.

[9] S. T. Roweis and L. K. Saul, “Nonlinear dimensionality reduction by locally linear embedding,” *Science*, pp. 2323–2326, dec 2000.

[10] R. G. Baraniuk, “Compressive sensing [lecture notes],” *IEEE Signal Process. Mag.*, vol. 24, no. 4, pp. 118–121, jul 2007.

[11] D. L. Donoho, “Compressed sensing,” *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, apr 2006.

[12] D. Malioutov, M. Cetin, and A. S. Willsky, "A sparse signal reconstruction perspective for source localization with sensor arrays," *IEEE Trans. Signal Process.*, vol. 53, no. 8, pp. 3010–3022, aug 2005.

[13] D. H. Johnson and D. E. Dudgeon, *Array Signal Processing: Concepts and Techniques*, Prentice Hall, Englewood Cliffs, NJ, 1993.

[14] J. Capon, "High-resolution frequency-wavenumber spectrum analysis," *Proceedings of the IEEE*, vol. 57, no. 8, pp. 1408–1418, aug 1969.

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

[16] G. Bienvenu and L. Kopp, "Adaptivity to background noise spatial coherence for high resolution passive methods," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP 1980)*, Denver, CO, apr 1980, vol. 5, pp. 307–310.

[17] P. Stoica and A. Nehorai, "MUSIC, maximum likelihood, and Cramér-Rao bound: Further results and comparisons," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 38, no. 12, pp. 2140–2150, dec 1990.

[18] D. Model and M. Zibulevsky, "Signal reconstruction in sensor arrays using sparse representations," *Signal Processing*, vol. 86, pp. 624–638, mar 2006.

[19] V. Cevher, M. Duarte, and R. G. Baraniuk, "Distributed target localization via spatial sparsity," in *European Signal Processing Conference (EUSIPCO 2008)*, Lausanne, Switzerland, aug 2008.

[20] V. Cevher, P. Indyk, C. Hegde, and R. G. Baraniuk, "Recovery of clustered sparse signals from compressive measurements," in *Int. Conf. Sampling Theory and Applications (SAMPTA 2009)*, Marseille, France, may 2009, pp. 18–22.

[21] S. Sen and A. Nehorai, "Sparsity-based multi-target tracking using OFDM radar," *IEEE Trans. Signal Process.*, vol. 59, pp. 1902–1906, Apr. 2011.

[22] S. Sen, G. Tang, and A. Nehorai, "Multiobjective optimization of OFDM radar waveform for target detection," *IEEE Trans. Signal Process.*, vol. 59, no. 2, pp. 639–652, feb 2011.

[23] M. M. Nikolić, G. Tang, A. Djordjević, and A. Nehorai, "Electromagnetic imaging using compressive sensing," submitted to *IEEE Trans. Signal Process.*

[24] A. Skodras, C. Christopoulos, and T. Ebrahimi, "The jpeg 2000 still image compression standard," *Signal Processing Magazine, IEEE*, vol. 18, no. 5, pp. 36–58, 2001.

[25] W. B. Pennebaker and J.L. Mitchell, *JPEG still image data compression standard*, Kluwer academic publishers, 1993.

[26] M. Aharon, M. Elad, and A. Bruckstein, "K-svd: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311 –4322, nov. 2006.

[27] J. Mairal, M. Elad, and G. Sapiro, "Sparse representation for color image restoration," *IEEE Trans. Image Process.*, vol. 17, no. 1, pp. 53 –69, jan. 2008.

[28] M. J. Fadili, J. L. Starck, and F. Murtagh, "Inpainting and zooming using sparse representations," *The Computer Journal*, vol. 52, no. 1, pp. 64, 2009.

[29] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T.S. Huang, and S. Yan, "Sparse representation for computer vision and pattern recognition," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1031 –1044, june 2010.

[30] E. J. Candes and Y. Plan, "Matrix completion with noise," *Proc. of the IEEE*, vol. 98, no. 6, pp. 925–936, jun 2010.

[31] E. J. Candès and B. Recht, "Exact matrix completion via convex optimization," *Found. Comput. Math.*, vol. 9, no. 6, pp. 717–772, 2009.

[32] E. J. Candès and T. Tao, "The power of convex relaxation: Near-optimal matrix completion," *IEEE Trans. Inf. Theory*, vol. 56, no. 5, pp. 2053 –2080, may 2010.

[33] Z. Liu and L. Vandenberghe, "Interior-point method for nuclear norm approximation with application to system identification," *SIAM Journal on Matrix Analysis and Applications*, vol. 31, no. 3, pp. 1235–1256, 2009.

[34] Y. Zhang, "A simple proof for recoverability of ℓ_1-minimization: go over or under?," Tech. Rep., Rice CAAM Department, 2005.

[35] D. L. Donoho and X. Huo, "Uncertainty principles and ideal atomic decomposition," *IEEE Trans. Inf. Theory*, vol. 47, no. 7, pp. 2845–2862, nov 2001.

[36] D. Donoho, "High-dimensional centrally-symmetric polytopes with neighborliness proportional to dimension," Technical report, Department of Statistics, Stanford University, 2004.

[37] M. Stojnic, F. Parvaresh, and B. Hassibi, "On the reconstruction of block-sparse signals with an optimal number of measurements," *IEEE Trans. Signal Process.*, vol. 57, no. 8, pp. 3075–3085, aug. 2009.

[38] B. Recht, W. Xu, and B. Hassibi, "Null space conditions and thresholds for rank minimization," *Mathematical Programming*, vol. 127, no. 1, pp. 175–202, 2011, 10.1007/s10107-010-0422-2.

[39] E. J. Candès, "The restricted isometry property and its implications for compressed sensing," *Compte Rendus de l'Academie des Sciences, Paris, Serie I*, vol. 346, pp. 589–592, 2008.

[40] A. Juditsky and A. Nemirovski, "On verifiable sufficient conditions for sparse signal recovery via ℓ_1 minimization," *Mathematical Programming*, vol. 127, no. 1, pp. 57–88, 2011.

[41] M. A. Sheikh, S. Sarvotham, O. Milenkovic, and R. G. Baraniuk, "DNA array decoding from nonlinear measurements by belief propagation," in *Proc. IEEE Workshop Statistical Signal Processing (SSP 2007)*, Madison, WI, aug 2007, pp. 215–219.

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

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

[44] C. Berge, *Topological Spaces*, Dover Publications, Mineola, NY, reprinted 1997 in paperback.

[45] G. H. Golub and C. F. Van Loan, *Matrix Computations*, Johns Hopkins University Press, Baltimore, MD, 3 edition, 1996.

[46] T. J. Ypma, "Historical development of the Newton-raphson method," *SIAM Review*, vol. 37, no. 4, pp. 531–?551, 1995.

[47] J. C. Butcher, *Numerical methods for ordinary differential equations*, John Wiley & Sons, 2008.

[48] M. Sniedovich, *Dynamic Programming: Foundations and Principles*, Taylor & Francis, 2010.

[49] A. Tarski, "A lattice-theoretical fix point theorem and its applications," *Pacific Journal of Mathematics*, vol. 5, pp. 285–309, 1955.

[50] S. Mendelson and N. Tomczak-Jaegermann, "A subgaussian embedding theorem," *Israel Journal of Mathematics*, pp. 349–364, mar 2008.

[51] S. Mendelson, A. Pajor, and N. Tomczak-Jaegermann, "Reconstruction and subgaussian operators in asymptotic geometric analysis," *Geometric And Functional Analysis*, pp. 1248–1282, nov 2007.

[52] M. Ledoux and M. Talagrand, *Probability in Banach spaces: Isoperimetry and processes*, vol. 23. of *Ergebnisse der Mathematik und ihrer Grenzgebiete*, 1991.

[53] K. Davidson and S. Szarek, “Local operator theory, random matrices and banach spaces,” in *Handbook of the Geometry of Banach Spaces*, W. B. Johnson and J. Lindenstrauss, Eds., vol. 1, pp. 317–366. North Holland, New York, NY, 2001.

[54] R. Vershynin, *Lecture notes on non-asymptotic random matrix theory*, 2007.

[55] R. Vershynin, “Introduction to the non-asymptotic analysis of random matrices,” in *Compressed Sensing: Theory and Applications*, Y. Eldar and G. Kutyniok, Eds. to appear.

[56] X. Fernique, *Régularité des trajectoires des fonctiones aléatoires gaussiennes*, Ecole d’Eté de Probabilités de St-Flour 1974, Lecture Notes in Mathematics 480, Springer-Verlag, 1975.

[57] M. Talagrand, “Regularity of Gaussian processes,” *Acta Math.*, vol. 159, pp. 99–149, 1987.

[58] M. Talagrand, *The generic chaining: upper and lower bounds of stochastic processes*, Springer, 2005.

[59] B. Carl and I. Stephani, *Entropy, compactness, and the approximation of operators*, Cambridge University Press, 1990.

[60] D. E. Edmunds and H. Triebel, *Function spaces, entropy numbers, differential operators*, Cambridge University Press, jan 1996.

[61] E. J. Candès and T. Tao, “Decoding by linear programming,” *IEEE Trans. Inf. Theory*, vol. 51, no. 12, pp. 4203–4215, dec 2005.

[62] E. G. Larsson and Y. Selen, “Linear regression with a sparse parameter vector,” *IEEE Trans. Signal Process.*, vol. 55, no. 2, pp. 451–460, feb 2007.

[63] D. L. Donoho and P. B. Stark, “Uncertainty principles and signal recovery,” *SIAM Journal on Applied Mathematics*, vol. 49, no. 3, pp. 906–931, 1989.

[64] F. Parvaresh, H. Vikalo, S. Misra, and B. Hassibi, “Recovering sparse signals using sparse measurement matrices in compressed DNA microarrays,” *IEEE J. Sel. Topics Signal Processing*, vol. 2, no. 3, pp. 275–285, jun 2008.

[65] M. A. Herman and T. Strohmer, “High-resolution radar via compressed sensing,” *IEEE Trans. Signal Process.*, vol. 57, no. 6, pp. 2275–2284, jun 2009.

[66] Z. Tian and G. B. Giannakis, “Compressed sensing for wideband cognitive radios,” in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP 2007)*, Honolulu, HI, apr 2007, pp. IV–1357–IV–1360.

[67] S. Mallat, *A Wavelet Tour of Signal Processing.*, Academic, San Diego, CA, 1998.

[68] I. F. Gorodnitsky and B. D. Rao, "Sparse signal reconstruction from limited data using FOCUSS: A re-weighted minimum norm algorithm," *IEEE Trans. Signal Process.*, vol. 45, no. 3, pp. 600–616, mar 1997.

[69] E. J. Candès and T. Tao, "The Dantzig selector: Statistical estimation when p is much larger than n," *Ann. Statist.*, vol. 35, pp. 2313–2351, 2007.

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

[71] W. Dai and O. Milenkovic, "Subspace pursuit for compressive sensing signal reconstruction," *IEEE Trans. Inf. Theory*, vol. 55, no. 5, pp. 2230–2249, may 2009.

[72] D. Needell and J. A. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Appl. Comput. Harmonic Anal.*, vol. 26, no. 3, pp. 301–321, may 2009.

[73] S. Gogineni and A. Nehorai, "Compressive sensing for MIMO radar with widely separated antennas," Submitted to *IEEE Trans. Signal Process.*

[74] S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM J. Sci. Comp.*, vol. 20, no. 1, pp. 33–61, 1998.

[75] R. Tibshirani, "Regression shrinkage and selection via LASSO," *J. Roy. Statist. Soc. Ser. B*, vol. 58, pp. 267–288, 1996.

[76] N. Meinshausen and B. Yu, "Lasso-type recovery of sparse representations for high-dimensional data," *Ann. Statist.*, vol. 37, pp. 246–270, 2009.

[77] M. R. Osborne, B. Presnell, and B. A. Turlach, "On the LASSO and its dual," *J. Comput. Graph. Statist.*, vol. 9, pp. 319–337, 2000.

[78] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Review*, vol. 43, no. 1, pp. 129–159, 2001.

[79] E. Candés and J. Romberg, "ℓ_1-magic: Recovery of sparse signals via convex programming," oct 2005.

[80] P. Bickel, Y. Ritov, and A. Tsybakov, "Simultaneous analysis of Lasso and Dantzig selector," *Annals of Statistics*, vol. 37, no. 4, pp. 1705–1732, 2009.

[81] P. J. Bickel, "Discussion of the Dantzig selector: statistical estimation when p is much larger than n, by e. j. candès and t. tao," *Annals of Stat.*, vol. 35, pp. 2352–2357, 2007.

[82] V. Kekatos and G. B. Giannakis, "Sparse Volterra and polynomial regression models: Recoverability and estimation," *Arxiv preprint arXiv:1103.0769*, 2011.

[83] R. DeVore R. Baraniuk, M. Davenport and M. B. Wakin, "A simple proof of the restricted isometry property for random matrices," *Constructive Approximation*, vol. 28, no. 3, pp. 253–263, 2008.

[84] H. Bodlaender, P. Gritzmann, V. Klee, and J. Leeuwen, "Computational complexity of norm-maximization," *Combinatorica*, vol. 10, pp. 203–225, 1990.

[85] M. Rudelson and R. Vershynin, "On sparse reconstruction from Fourier and Gaussian measurements," *Communications on Pure and Applied Mathematics*, vol. 61, pp. 1025–1045, 2008.

[86] H. Rauhut, "Compressive sensing and structured random matrices," *Theoretical Foundations and Numerical Methods for Sparse Recovery*, vol. 9, pp. 1–92, 2010.

[87] G. Tang and A. Nehorai, "Performance analysis of sparse recovery based on constrained minimal singular values," submitted for publication, *ArXiv e-prints* available.

[88] M. Mishali and Y. C. Eldar, "Blind multi-band signal reconstruction: Compressed sensing for analog signals," *IEEE Trans. Signal Process.*, vol. 57, no. 3, pp. 993–1009, Mar. 2009.

[89] H. Liu, J. Zhang, X. Jiang, and J. Liu, "The group dantzig selector," in *Inter. Conf. Artificial Intelligence and Statistics (AISTATS 2010)*, Sardinia, Italy, May 2010.

[90] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 68, no. 1, pp. 49–67, 2006.

[91] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming (web page and software)," jun 2009.

[92] Y. Nesterov, "Smoothing technique and its applications in semidefinite optimization," *Math. Program.*, vol. 110, pp. 245–259, March 2007.

[93] G. Tang and A. Nehorai, "Verifiable and computable ℓ_∞ performance evaluation of ℓ_1 sparse signal recovery," 2011.

[94] L. El Ghaoui and P. Gahinet, "Rank minimization under LMI constraints: A framework for output feedback problems," in *Proc. European Control Conf.* 1993.

[95] M. Fazel, H. Hindi, and S. Boyd, "A rank minimization heuristic with application to minimum order system approximation," in *Proc. American Control Conf.* 2001.

[96] D. Gross, Y. Liu, S. T. Flammia, S. Becker, and J. Eisert, "Quantum state tomography via compressed sensing," *Phys. Rev. Lett.*, vol. 105, no. 15, pp. 150401–150404, oct 2010.

[97] R. Basri and D. W. Jacobs, "Lambertian reflectance and linear subspaces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 2, pp. 218 – 233, feb 2003.

[98] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?," *J. ACM*, vol. 58, no. 3, pp. 1–37, jun 2011.

[99] N. Linial, E. London, and Y. Rabinovich, "The geometry of graphs and some of its algorithmic applications," *Combinatorica*, vol. 15, pp. 215–245, 1995.

[100] B. Recht, M. Fazel, and P. A. Parrilo, "Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization," *SIAM Review*, vol. 52, no. 3, pp. 471–501, 2010.

[101] M. Fazel, *Matrix Rank Minimization with Applications*, Ph.D. thesis, Stanford University, 2002.

[102] S. Ma, D. Goldfarb, and L. Chen, "Fixed point and Bregman iterative methods for matrix rank minimization," *Mathematical Programming*, vol. 120, no. 2, pp. 1–33, 2009.

[103] G. Tang and A. Nehorai, "The ℓ_1-constrained minimal singular value: a computable quantification of the stability of sparse signal reconstruction," submitted to *IEEE Trans. Information Theory.*

[104] A. Juditsky, F. K. Karzan, and A. S. Nemirovski, "On low rank matrix approximations with applications to synthesis problem in compressed sensing," *ArXiv e-prints*, 2010.

[105] V. Chandrasekaran, B Recht, P. A. Parrilo, and A. Willsky, "The convex geometry of linear inverse problems," Submitted for publication, 2010.

[106] S. Lansel, M. Parmar, and B. A. Wandell, "Dictionaries for sparse representation and recovery of reflectances," in *Proc. SPIE*, vol. 72460.

Vita

Gongguo Tang

Date of Birth	June 27, 1981
Place of Birth	Shandong, China
Degrees	Ph.D. Electrical Engineering, August 2011 M.S. Control and Systems Science, July 2006 B.S. Mathematics, July 2003
Professional Societies	Institute of Electrical and Electronics Engineers Society for Industrial and Applied Mathematics

Publications

Journal Papers:

1. **G. Tang** and A. Nehorai, "Fixed-point iteration and semidefinite programming for computable performance analysis of block-sparsity recovery", submitted for publication.
2. Y. Peng, **G. Tang**, and A. Nehorai, "Sparsity-enforced regression based on over-complete dictionary", submitted for publication.
3. **G. Tang** and A. Nehorai, "Verifiable and computable ℓ_∞ performance evaluation of ℓ_1 sparse signal recovery," submitted for publication.
4. **G. Tang** and A. Nehorai, "Performance analysis of sparse signal recovery based on constrained minimal singular values," in revision for *IEEE Trans. Signal Processing.*
5. **G. Tang** and A. Nehorai, "Constrained Cramr-Rao bound for robust principal component analysis," to appear in *IEEE Trans. Signal Processing.*
6. **G. Tang** and A. Nehorai, "Lower bounds on mean-squared error for low-rank matrix reconstruction," to appear in for *IEEE Trans. Signal Processing.*
7. **G. Tang** and A. Nehorai, "The stability of low-rank matrix reconstruction: a constrained singular value view," in revision for *IEEE Trans. Information Theory.*

8. **G. Tang** and A. Nehorai, "Performance analysis for sparse support recovery," *IEEE Trans. Information Theory*, vol. 56, no. 3, pp. 1383-1399, Mar. 2010.
9. S. Sen, **G. Tang**, and A. Nehorai, "Multi-objective optimized OFDM radar waveform for target detection in multipath scenarios," *IEEE Trans. Signal Processing*, vol. 59, no. 2, pp. 639-652, Feb. 2011.
10. M. M. Nikolic, **G. Tang**, A. Djordjevic, and A. Nehorai, "Electromagnetic imaging using compressive sensing," in revision for *IEEE Trans. Signal Processing.*
11. **G. Tang** and L. Guo, "Convergence of a class of multi-agent systems in probabilistic framework," *Journal of Systems Science and Complexity*, vol. 20, no. 2, pp. 173-197, Jun. 2007.

Conference Papers:

1. P. Yang, **G. Tang**, and A. Nehorai, "Sparsity enforced regression based on overcomplete dictionary," submitted to *International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP 2011).*
2. **G. Tang** and A. Nehorai, "Verifiable and computable ℓ_∞ performance evaluation of ℓ_1 sparse signal recovery," in *Proc. 45th Annual Conference on Information Sciences and Systems (CISS2011).*
3. **G. Tang** and A. Nehorai, "Robust principal component analysis based on low-rank and block-sparse matrix decomposition," in *Proc. 45th Annual Conference on Information Sciences and Systems (CISS2011).*
4. S. Sen, **G. Tang**, and A. Nehorai, "Sparsity-based estimation for target detection in multipath scenarios," *RadarCon'11.*
5. **G. Tang** and A. Nehorai, "The stability of low-rank matrix reconstruction: a constrained singular value perspective," in *Proc. 48th Allerton Conf. on Communication, Control, and Computing*, Monticello, IL, Sep. 2010, pp. 1745-1751.
6. **G. Tang** and A. Nehorai, "Computable quantification of the stability of sparse signal reconstruction," in *Proc. Asilomar Conf. on Signals, Systems, and Computers* (2010).

7. S. Sen, **G. Tang**, and A. Nehorai, “Multi-objective optimized OFDM radar waveform for target detection in multipath scenarios,” in *Proc. Asilomar Conf. on Signals, Systems, and Computers* (2010).
8. M. M. Nikolic, **G. Tang**, and A. Nehorai, “Electromagnetic imaging using compressive sensing,” in *Proc. 48th Allerton Conf. Communication, Control and Computing*, Monticello, IL, Sep. 2010, pp. 1737-1741.
9. M. M. Nikolic, **G. Tang**, and A. Nehorai, “3D electromagnetic imaging using compressive sensing,” in *IEEE Sensor Array and Multichannel Signal Process. Workshop (SAM 2010)*, Jerusalem, Israel, Oct. 2010, pp. 37-40.
10. **G. Tang** and A. Nehorai, “Support recovery for source localization based on overcomplete signal representation,” in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP 2010)*, Dallas, TX, Mar. 2010, pp. 2730-2733.
11. **G. Tang** and A. Nehorai, “Performance analysis for support recovery with joint sparsity constraints,” in *Proc. 47th Allerton Conf. Communication, Control, and Computing*, Monticello, IL, Sep. 2009, pp. 258-264.
12. **G. Tang** and A. Nehorai, “Performance bounds on sparse-pattern recovery for multiple signals with a common support,” in *Proc. Defense Applications Signal Processing Workshop (DASP 2009)*, Lihue, Hawaii, Sep. 2009.
13. **G. Tang** and L. Guo, “Convergence of a class of multi-agent systems in probabilistic framework,” in *Proc. 46th IEEE Conf. on Decision and Control (CDC 2007)*, New Orleans, LA, Dec. 2007, pp. 318-323.
14. **G. Tang** and L. Guo, “Convergence analysis of linearized Vicsek’s model,” in *Proc. 25th Chinese Control Conference (CCC 2006)*, Harbin, China, Aug. 2006, pp. 379-382.
15. **G. Tang** and L. Guo, “Adaptive stabilization of unknown linear systems under communication constraints,” in *Proc. 24th Chinese Control Conference (CCC 2005)*, Guangzhou, China, Jul. 2005, pp. 668-672.

August 2011

Computable Performance Analysis, Tang, Ph.D. 2011