Imperial College London
Department of Electrical and Electronic Engineering

Fast Proximal Gradient Methods for Spectral Compressed Sensing via Multi-block Hankel Matrices

Xi Yao

Supervised by Dr. Wei Dai

The Thesis Submitted to the Electrical and Electronic Engineering Department
in Fulfillment of the Requirements for the Degree of
Doctor of Philosophy, 2024

Statement of Originality

I hereby declare that this thesis and the work reported herein is composed by and originated entirely from me under the guidance of my PhD supervisor Dr. Wei Dai. Information derived from the published and unpublished work of others has been acknowledged in the text and references are given in the list of sources.

Xi Yao

Copyright Declaration

Abstract

This thesis investigates the recovery of a spectrally sparse signal (SSS) from partially observed, noisy data. Traditional compressed sensing methods encounter a *basis mismatch issue* due to the finite discrete dictionary. To address this, recent literature introduces grid-free approaches that exploit frequency sparsity in a continuous manner. Among these, the enhanced Hankel matrix method stands out for its improved resolution. Moreover, the algorithm involves the Hankel matrix as a whole, ensuring efficient storage and computational processes. Despite these advancements, the method shares a common drawback with Cadzow's method: reconstruction accuracy is compromised by the repeated elements in the Hankel structure. This issue is critical in scenarios where precise signal reconstruction is essential. Additionally, these methods face challenges in convergence speed, particularly when first-order methods are employed, as they tend to converge slowly in cases of low sampling ratios or significant noise.

To address these issues in spectral compressed sensing, this thesis introduces a new nonconvex optimization framework. This framework measures reconstruction error in the signal space rather than in the lifted Hankel space and incorporates an adjustable Hankel constraint parameter tailored to noise levels. To improve the slow convergence and recovery ability of standard proximal gradient (PG) methods, three advanced PG-based algorithms are proposed: low-rank projected proximal, Hankel projected proximal, and Hessian proximal gradient. These algorithms are meticulously designed to utilize the intrinsic low-rank and Hankel structures of the problem, enhancing computational efficiency. This efficiency is supported by a Julia package, available at `https://github.com/xiyao65/multiblockHankelMatrices.jl`. Numerical simulations demonstrate a significant improvement in both efficiency and recovery accuracy. This enhancement is particularly notable in scenarios with substantial noise or low sampling ratios, underscoring the methods' robustness and applicability for large-scale SSS recovery tasks.

Acknowledgements

First, I would like to express my deep appreciation to my PhD advisor, Dr. Wei Dai, for his outstanding mentorship, support, and guidance throughout my doctoral studies. His dynamic approach to teaching, profound academic counsel, thorough review of papers, and consistent availability for meetings have been crucial to my progress. I am also profoundly grateful to the members of my peer group for their technical advice and daily friendship. Additionally, my heartfelt thanks go to my friends Yang, Yumeng, Yunhao, Shixiong, and everyone in the CSP group for making my experience in London both enriching and unforgettable.

My heartfelt thanks are extended to my parents and family for their constant support. Their love, comprehension, and endorsement have been the cornerstones of my resilience. Even with the considerable geographical distance separating us, their empathetic grasp of my thoughts and choices, along with their reassuring conversations during video chats, have consistently been a source of encouragement during times of uncertainty.

Lastly, my heartfelt thanks go to my wife, Jenny, for her patience, and trust. Her family offers me plenty of hotpot and barbeque to take several warm winters in London. Our cat, Wofly, a shy and beautiful ragdoll, brings me a lot of joy with his companionship.

Nomenclature

Roman Symbols:

a	scalar
$\boldsymbol{a}$	vector
$\boldsymbol{A}$	matrix
$\boldsymbol{I}_{n\times n}$	n-by-n identity matrix
$\mathbb{R}$, $\mathbb{R}^n$, $\mathbb{R}^{m\times n}$	set of real numbers, vectors, and matrices
$\mathbb{C}$, $\mathbb{C}^n$, $\mathbb{C}^{m\times n}$	set of complex numbers, vectors, and matrices
a_{ij}, $\boldsymbol{A}(i,j)$	(i,j) entry of a matrix
$\|\boldsymbol{a}\|_p$	p-norm of a vector
$\|\boldsymbol{A}\|_p$, $\|\boldsymbol{A}\|_F$	p-norm and Frobenius norm of a matrix
$\|\boldsymbol{A}\|^{\mathsf{T}}$, $\|\boldsymbol{A}\|^{\mathsf{H}}$	transpose and conjugate transpose of a matrix
$\mathrm{rank}(\boldsymbol{A})$, $\mathrm{trace}(\boldsymbol{A})$	rank and trace of a matrix
$\mathrm{vec}(\boldsymbol{A})$	vectorization of a matrix
$\langle\cdot,\cdot\rangle$	inner product

Calligraphic letters:

$\mathcal{H}$	Hankel operator
$\mathcal{H}^*$	adjoint operator of Hankel operator
$\mathcal{H}^\dagger$	Penrose Pseudoinverse operator of Hankel operator
$\mathcal{P}_\Omega$	sampling operator corresponding to the index set Ω
$\mathcal{T}$	truncated SVD operator

$\mathcal{I}$	identity mapping

List of Abbreviations:

ANM	Atomic Norm Minimization
CS	Compressed Sensing
EMaC	Enhanced Matrix Completion
FFT	Fast Fourier Transform
FIHT	Fast Iterative Hard Thresholding
HPG	Hessian Proximal Gradient
HPPG	Hankel Projected Proximal Gradient
ISTA	Iterative Shrinkage-thresholding Algorithm
LPPG	Low-rank Projected Proximal Gradient
MBH	Multi-block Hankel
MLR	Maximum Likelihood Estimation
MMV	Multiple Measurement Vector
MRI	Magnetic Resonance Imaging
NMR	Nuclear Magnetic Resonance
NMSE	Normalized Mean Squared Error
PGD	Proximal Gradient Descent
PG	Proximal Gradient
SCS	Spectral Compressed Sensing
SDP	Semidefinite Programming
SSS	Spectrally Sparse Signal
SVD	Singular Value Decomposition
SVT	Singular Value Thresholding

Remark 1. *To reduce notational clutter, the Hankel operation $\mathcal{H}(\cdot)$ on $\mathcal{X}$ is shorthanded as $\mathcal{H}\boldsymbol{x} := \mathcal{H}(\boldsymbol{x})$ for all $\boldsymbol{x} \in \mathcal{X}$. The same rule applies to $\mathcal{P}_\Omega(\cdot)$, $\mathcal{H}^*(\cdot)$, etc.*

Contents

List of Tables

List of Figures

List of Algorithms

Part I

Introduction

Chapter 1

Background and Contributions

1.1 Motivations and Objectives

A large class of practical applications involves signals that can be effectively modeled or approximated as a weighted superposition of several complex sinusoids, termed spectrally sparse signals (SSSs). The characteristic feature of these signals is their spectral sparsity, indicated by a limited number of spectral components. However, the data acquisition devices are often limited by physical and hardware constraints, precluding full sampling with the desired resolution. It is necessary to develop spectral compressed sensing (SCS) techniques [8, 9] for efficient recovery of undermined frequencies from a subset of regularly time-spaced samples. The SCS plays a pivotal role in various fields. For example, in medical imaging, it contributes to enhanced resolution and reduced scan times [10], while in radar imaging, SCS facilitates more accurate object detection [11]. Similarly, in fluorescence microscopy [12], wireless communication channel estimation [13], and time series prediction [14], the SSSs are widely used. Additionally, these signals can exhibit damping, a decline in amplitude over time. This decay, represented as a sum of damped complex sinusoids, is particularly evident in applications such as biological nuclear magnetic resonance (NMR) spectroscopy [15] and linear recurrent time series [14]. The decaying signal presents additional recovery challenges, such as increased sensing complexity and noise sensitivity. Addressing these challenges is crucial, underscoring the importance of advancing

SCS techniques, a primary focus of this thesis.

In numerous scenarios, the successful recovery of an object is feasible with a sample count significantly lower than its ambient dimension, due to the object's sparse representation within the transform domain. The foundational concept of sparse methods in compressed sensing (CS), with historical roots extending to the previous century, was notably advanced by early works such as Gorodnitsky and Rao [16]. Subsequent developments in CS, particularly through tractable approaches utilizing convex surrogate methodologies [8, 17], have not been restricted by prior model order knowledge and are recognized for their robustness against noise. The evolution of CS has witnessed the advent of diverse algorithmic strategies based on finite discrete dictionaries. Among these, iterative thresholding [18], subspace pursuit [19], and the CoSaMP algorithm [20] are noteworthy. However, despite these advancements, CS continues to face a fundamental limitation due to its reliance on sparse representation within a finite discrete dictionary. This limitation becomes particularly pronounced in scenarios where actual parameters are aligned with a continuous dictionary. The phenomenon of frequency leakage, a direct consequence of this misalignment, arises when actual frequencies do not align with the predetermined grid. This discrepancy poses notable challenges in accurately capturing the true essence of signals, as explored in studies [21, 22].

The SCS for continuous frequencies marks a significant shift from traditional CS techniques [8] by not constraining frequencies to a predetermined grid. The SCS problem is intrinsically linked to line spectral estimation, or harmonic retrieval, which aims to extract the underlying frequencies of an object from its full-time-domain samples. This methodology finds applications across various signal processing domains, including radar localization systems [23], array imaging systems [24], and wireless channel sensing [25, 26]. Accurate estimation of the time-domain representation of an object enables the identification of underlying frequencies through harmonic super-resolution techniques. The field of line spectral estimation has a rich history of research, encompassing a wide range of approaches, as detailed in [27]. A prominent nonparametric method in this context is the fast Fourier transform (FFT) [28]. However, the quest for higher accuracy and resolution, driven by increasing computing power, has led to the development of

parametric methods like maximum likelihood estimation (MLE) in which the main difficulty comes from the nonlinearity and nonconvexity concerning the frequency parameters. To address these complexities, subspace-based methods such as MUSIC [29], Prony's method [30], and the matrix pencil approach [31] have been utilized. These techniques leverage the shift-invariance characteristic of harmonic structures, where consecutive segments of time-domain samples reside in the same subspace, regardless of the segment's starting point. Nonetheless, these methods face limitations in handling incomplete time-domain samples and often exhibit sensitivity to noise and outliers [32], which poses significant challenges in practical applications.

In contrast, when faced with missing data, the task evolves into super-resolution CS [33, 34], which is focused on the finest resolution of estimated parameters despite incomplete datasets. This approach, by harnessing spectral sparsity, presents an optimization framework for signal recovery that circumvents frequency discretization issues inherent in on-grid methods. The groundbreaking works [33, 34] established a robust theoretical and algorithmic foundation, introducing a convex optimization approach to handle continuous-valued frequencies. To date, several convex, gridless sparse methods have also been developed, including atomic norm (or equivalently total variation norm) methods [33, 35–39], enhanced matrix completion (EMaC) [2], and covariance fitting approaches [40–42]. These methods aim to optimize a structured low-rank matrix explicitly or implicitly, with its low-rankness attributed to spectral sparsity. Specifically, atomic norm and covariance fitting methods are associated with a low-rank positive semidefinite (PSD) Toeplitz matrix, whereas EMaC is based on a low-rank Hankel matrix. In parallel, MLE approaches [43, 44] tackle frequency estimation in a nonconvex manner. Unfortunately, these PSD and MLE methods are not feasible for large-scale CS due to their high computational complexity [45]. Furthermore, extending these methods to recover damped exponential signals remains an unresolved challenge in the field.

To adapt SCS to large-scale general SSSs, including damped and multi-dimensional signals, our focus is on the EMaC approach for two main reasons: 1) EMaC leverages the Vandermonde decomposition to extend the atomic set provided by atomic norm minimization [35], making it applicable to damped signals [46]; 2) It formulates the reconstruction of SSSs as a low-rank

matrix completion problem by arranging data into a multi-fold Hankel structure, enhancing both storage and computational efficiency. After recovering the corrupted and subsampled SSSs in the time domain, which means the matrix approximation, the line spectral estimation [47] methods can be applied to estimate the original frequencies and coefficients. However, structured low-rank approximation problems, such as those encountered in EMaC, are known to be NP-hard global optimization challenges [48]. Specifically, the number of stationary points in Hankel low-rank approximation problems increases polynomially with size n and exponentially with rank r [49]. Inspired by low-rank completion techniques [50], EMaC addresses the Hankel matrix completion problem via nuclear norm minimization, posing it as a convex problem with recovery guarantees. This approach has gained popularity for large-scale problems, particularly due to the computational efficiency offered by the compact Hankel structure, which supports efficient singular value decomposition (SVD) [51]. An alternative approach involves the Burer-Monteiro heuristic [52, 53], where low-rank matrices are represented through a bilinear outer product, circumventing explicit matrix SVD. Nevertheless, it is important to note that worst-case analyses [45] indicate the potential failure of these convex optimizations even with a single missing element. Furthermore, nuclear norm-based approaches generally necessitate the full SVD of the enhanced matrix, which can be prohibitive for large-scale applications. Consequently, various non-convex methods [14, 54–58] employing hard rank constraints have been developed to address low-rank Hankel matrix completion problems more directly.

While EMaC-based algorithms are capable of handling general SSSs, they encounter two primary limitations. First, the convergence to critical points remains challenging despite the use of truncated SVD to reduce computational complexity. This difficulty arises from both the number of iterations and the computation complexity per iteration. The Cadzow's algorithm [14, 54] employs alternating projections between Hankel matrices and low-rank matrices. An attempt to achieve quadratic convergence through an additional Newton-type step [59], based on common regularity intersection conditions [60, 61] of Cadzow's algorithm, has been hindered by high computational costs. Acceleration methods inspired by Riemannian optimization [62] and direct optimization on component matrices [55, 56] have been explored, with fast iterative hard thresholding (FIHT) [57] and projected gradient descent (PGD) [56] showing efficiency and

convergence guarantees. However, these algorithms often exhibit slow convergence, particularly in cases of low sampling or high noise levels. Second, EMaC-based algorithms demonstrate sensitivity to noise. This issue stems from the formulation of the objective SSS in enhanced matrix norm, either implicitly or explicitly, rather than in its intrinsic vector form. Such a formulation leads to a biased weighted norm due to repeated elements in the induced Hankel matrix [14, 58, 63, 64], complicating the accurate recovery of signals in noisy environments.

To summarize, the application of enhanced low-rank Hankel matrix approximation in the context of large-scale, multi-dimensional, and possibly damped SSS recovery encounters three primary challenges:

1. **Biased Norm:** The presence of a biased weighted norm in the induced Hankel matrix leads to amplified bias in signal estimation, particularly in scenarios with large noise levels and signal sizes.

2. **Computational Complexity per Iteration:** There are significant computational costs involved in solving the structured low-rank approximation problem or accelerating the convergence speed. This computational challenge becomes more pronounced as signal size increases, leading to increased processing time and resource requirements.

3. **Slow Convergence:** The convergence speed is often hindered by the condition number of the Hessian matrix associated with the quadratic terms in the objective function. This condition number is influenced by the signal size and sampling ratio, contributing to slower convergence rates.

1.2 Original Contributions and Outline of the Thesis

To address the challenges previously identified in the recovery of SSSs, this thesis introduces fast proximal gradient-based approaches, utilizing the low-rank Hankel matrix structure and an unweighted norm to enhance signal recovery performance. The main contributions of this thesis are summarized as follows:

1. **Efficient Implementations:** The Hankel structure is crucial for fast computations. To utilize this, a Hankel matrix package has been developed in Julia, a high-level, high-performance dynamic programming language for numerical computing. This package includes essential functions such as compact Hankel matrix construction, fast matrix-vector multiplications, truncated SVD, and the adjoint operator of Hankel mapping on low-rank matrices. It supports efficient implementation of the proposed algorithms and is extendable to multilevel Hankel matrices. The package is available at `https://github.com/xiyao65/multiblockHankelMatrices.jl`.

2. **Accurate Formulations:** Our approach aligns more accurately with the original problem model in two key aspects. Unlike conventional Hankel-based algorithms that represent the SSS in an enhanced matrix form, our method utilizes a vector form representation, thereby reducing bias from different weights induced by the Hankel operator [14, 58]. Additionally, we incorporate a Hankel enforcement parameter in our unconstrained optimization framework to adapt to various noise levels, enhancing reconstruction accuracy in noisy environments.

3. **Fast and Convergent Proximal Gradient (PG) Algorithms for Different Scenarios:** This thesis introduces three distinct PG-based algorithms that converge toward critical points without the need for initialization constraints, demonstrating exceptional recovery capabilities in a nonconvex setting. These include the low-rank projected proximal gradient (LPPG), Hankel projected proximal gradient (HPPG), and Hessian proximal gradient (HPG) algorithms. The standard PG approach [6], ensures a monotonically decreasing function with a fixed step size related to the length of signal. However, it often suffers from slow convergence, particularly in challenging scenarios characterized by large-scale and outliers. To mitigate this slow issue, the LPPG algorithm has been developed to accelerate convergence with a larger decrease in the objective function, thereby enhancing both the speed and accuracy of reconstruction. Additionally, the HPPG algorithm is specifically designed for low sampling situations, enabling improved reconstruction accuracy and faster convergence by utilizing larger step sizes through the exploitation of the Hankel structure. Lastly, the HPG algorithm is introduced to further expedite convergence by leveraging the

Hessian matrix of the quadratic terms in the objective function. This second-order method is particularly robust against large condition numbers of the Hessian matrix, which are influenced by the Hankel enforcement parameter, signal size, and sampling ratio, making it highly effective in complex recovery tasks. All these three algorithms exploit the low-rank Hankel structure, thereby enhancing computational efficiency.

The remainder of this thesis is divided into the following three main parts:

1. **Background:** In Chapter 2, a review of the field of SSS recovery is provided. This chapter covers two typical traditional gridless SCS methods, atomic norm minimization and enhanced matrix completion approaches. It also delves into formulations related to low-rank Hankelization and critically assesses the limitations of existing algorithms, setting the stage for the introduction of novel methodologies.

2. **Methodologies:** This section elaborates on the implementation and efficacy of the proposed algorithms.

   - Chapter 3 is dedicated to the fast computations of multi-block Hankel matrices, a foundational aspect for the efficient implementation of our algorithms.
   - Chapter 4 introduces the LPPG method, tailored to tackle the challenges of large-scale data and significant noise in SSS recovery. This chapter serves as the foundation for the SSS recovery problem via the PG-based methods, detailing components such as the optimization framework's objective function, a modified PG method for consistent step sizes, and a convergence analysis of the nonconvex PG-based algorithm.
   - In Chapter 5, the HPPG approach is presented. This method addresses the computational complexities identified in LPPG and is specifically optimized for low-samplingng scenarios to enhance reconstruction accuracy and speed up convergence.
   - Chapter 6 discusses the HPG, a second-order method designed to improve convergence rates while ensuring accurate reconstruction. This method is particularly effective for reconstructing SSSs from heavily noisy data and with uncertain model orders.

3. **Conclusions:** Chapter 7 concludes the thesis, providing a comparative evaluation of the three proposed algorithms. This chapter highlights their distinct advantages and explores potential directions for future research in the field.

These proposed PG-based approaches systematically address the challenges in SSS recovery, offering innovative solutions and advancing the field.

1.3 Publications

The content of this thesis mainly included three published work and one submitted journal work.

Yao, Xi, and Wei Dai. "A Projected Proximal Gradient Method for Efficient Recovery of Spectrally Sparse Signals." 2023 31st European Signal Processing Conference (EUSIPCO). IEEE, 2023. (Related to Chapter 4).

Yao, Xi, and Wei Dai. "A Preconditioned Hessian Proximal Algorithm for Spectral Compressed Sensing." 2023 IEEE 9th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP). IEEE, 2023. (Related to Chapter 6).

Yao, Xi, and Wei Dai. "Accelerated Recovery of Spectrally Sparse Signals Via Modified Proximal Gradient in Hankel Space." 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2024.(Related to Chapter 5).

Yao, X., Wei, D. "A Low-rank Projected Proximal Gradient Method for Spectral Compressed Sensing." arXiv:2405.07739. (Related to Chapter 4).

Chapter 2

Preliminaries

In this chapter, we review various aspects of *super-resolution* SCS and the signal model related to SCS. The chapter is structured as follows:

1. **Background on *Super-resolution* CS:** Section 2.1 delves into the foundational concepts of *super-resolution* and the spectrally sparse signal model, providing a detailed background for the subsequent discussion.

2. **Fundamental Approaches to Gridless SCS:**

    - Section 2.2 discusses the atomic norm minimization (ANM) approach, detailing its principles and applications in the context of *super-resolution* of SCS.
    - Section 2.3 focuses on the enhanced Hankel matrix completion approach, outlining its methodology and advantages.

3. **Nonconvex Nonsmooth Hankelization Formulation:** Section 2.4 presents the nonconvex nonsmooth Hankelization formulation via the lifting technique, a critical component of our proposed approach to SCS.

4. **Motivations Behind Our Algorithms:** In Section 2.5, we explain the underlying motivations and the theoretical underpinnings that drive the development of our algorithms.

5. **Advantages of Proximal Gradient Based Methodologies:** Finally, Section 2.6 summarizes the key advantages and distinctive features of our proximal gradient (PG) based methodologies, highlighting their contributions to the field of SCS.

This chapter aims to provide a thorough understanding of the current state of *super-resolution* of SCS techniques and to articulate the rationale and benefits of the novel approaches developed in this thesis.

2.1 *Super-resolution* and Signal Model

2.1.1 Gridless Spectral Compressed Sensing

In a variety of sensing applications, the signal under study is typically formulated as a linear combination of either translated or modulated versions of a standard model, like a point spread function (PSF). A major challenge here is to accurately estimate parameters related to translation or modulation—such as delays, positions, or Doppler shifts—from data that is often noisy. The resolution capability of signals captured by sensing or imaging devices is inherently linked to the limitations of these devices.

In contrast, the resolution required for accurate parameter estimation often surpasses what is achievable with conventional sensing or imaging methods, presenting a challenge known as "*super-resolution*". In the work [33], the notion *super-resolution* was first introduced to describe the ability of resolving the true parameters with infinite precision [65]. The development of *super-resolution* algorithms has been a prominent pursuit within the field of signal processing, particularly pertinent when dealing with fully sampled data, and has a rich history of significant contributions [27]. One of the earliest techniques, dating back to 1795, is de Prony's root-finding method [30], which has undergone numerous enhancements over time, particularly in noise management [66]. Since the 1980s, the evolution of subspace methods like MUSIC [29], ESPRIT [67], and matrix pencil [31] marked significant advancements in the field. These

algorithms possess the capability for *super-resolution*, meaning they can discern parameters at a finer resolution, especially in conditions of low noise. However, despite their advanced resolution capabilities, these methods are highly sensitive to noise and outliers, and their performance is significantly degraded in the presence of missing data.

The highlighted limitations in traditional *super-resolution* methods called for the development of *super-resolution* CS techniques based on convex relaxations [2, 33, 35]. These approaches have shown to match the performance of harmonic *super-resolution* methods while also displaying increased resilience against factors like noise, missing data, and outliers. A significant advantage of these methods is that they avoid the need for discretizing delays over a finite grid, overcoming a frequent constraint of conventional CS approaches [21, 68]. Once the the original data is recovered the harmonic *super-resolution* methods could be adopted easily. Consequently, this thesis centers on gridless *super-resolution* approaches of reconstruction, which are becoming increasingly prominent in fields such as signal processing, applied mathematics, and optimization.

The appeal of convex optimization, integral to these approaches, stems from its tractability and solid theoretical foundation. This optimization technique stands out for several reasons: Firstly, it is underpinned by strong theoretical guarantees, assuring its efficacy even in environments with noisy or corrupt data. Secondly, its flexible nature permits the integration of prior knowledge into the convex program, thereby widening its applicability well beyond that of traditional methods. Lastly, recent progress in large-scale convex optimization has led to the creation of efficient computational solvers, significantly enhancing the practicality of these methods for real-world application scenarios.

2.1.2 Notation

In this text, vectors and matrices are represented by bold lowercase and uppercase letters, respectively, such as $\boldsymbol{x}$ for vectors and $\boldsymbol{X}$ for matrices. The transpose and Hermitian of a matrix $\boldsymbol{X}$ are denoted as $\boldsymbol{X}^{\mathsf{T}}$ and $\boldsymbol{X}^{\mathsf{H}}$, respectively. The symbols $\|\boldsymbol{x}\|_2$ and $\|\boldsymbol{X}\|_F$ indicate the l_2 norm of a vector and the Frobenius norm of a matrix, respectively. The diagonal matrix formed

from the vector $\boldsymbol{x}$ is represented as $\mathrm{diag}(\boldsymbol{x})$. It is important to note that a Hankel matrix is characterized by having constant skew diagonals.

Consider $\mathcal{H}$ as the linear operator that maps a vector $\boldsymbol{x} = [x_0, x_1, \cdots, x_{n-1}]^\mathsf{T} \in \mathbb{C}^n$ to a Hankel matrix in $\mathbb{C}^{p\times q}$, where $p + q = n + 1$ and $p \geq q$. The element of the enhanced Hankel matrix by the Hankel operator $\mathcal{H}\boldsymbol{x}[i, j]$ corresponds to $\boldsymbol{x}_{i+j}$:

$$\boldsymbol{X}^e = \mathcal{H}\boldsymbol{x} = \begin{bmatrix} x_0 & x_1 & \dots & x_{q-1} \\ x_1 & x_2 & \dots & x_q \\ \vdots & & \vdots & \\ x_{p-1} & x_p & \dots & x_{n-1} \end{bmatrix} \in \mathbb{C}^{p\times q}, \tag{2.1}$$

with indexing starting from 0.

Denote $\mathcal{H}^*$ as the adjoint operator of $\mathcal{H}$, a linear operator mapping $p\times q$ matrices to n-dimensional vectors. For any matrix $\boldsymbol{X}^e \in \mathbb{C}^{p\times q}$, $\mathcal{H}^*\boldsymbol{X}^e$ aggregates the elements of $\boldsymbol{X}^e$ along skew diagonals: $\left\{\sum_{i+j=a} \boldsymbol{X}^e[i, j]\right\}_{a=0}^{n-1}$. Consequently, $\mathcal{W} \coloneqq \mathcal{H}^*\mathcal{H}$ forms a diagonal matrix equivalent to $\mathrm{diag}(\boldsymbol{w})$, with $\boldsymbol{w} = [1, \dots, q-1, \underbrace{q, \cdots, q}_{p-q+1}, q-1, \dots, 1]^\mathsf{T}$. Additionally, the left inverse operator of $\mathcal{H}$, denoted as $\mathcal{H}^\dagger$, retrieves $\boldsymbol{x}$ from matrix space, defined as $\mathcal{H}^\dagger = \mathcal{W}^{-1}\mathcal{H}^*$. Let $\Omega \subset \{1, \dots, n\}$ denote the set of indices corresponding to the observed entries with $|\Omega| = m \leq n$. Let $\mathcal{P}_\Omega$ be the associated sampling operator of the original signal $\boldsymbol{x}$, i.e., $\mathcal{P}_\Omega\boldsymbol{x} = \sum_{k\in\Omega}\langle\boldsymbol{x}, \boldsymbol{e}_k\rangle\boldsymbol{e}_k$, where $\boldsymbol{e}_k$ be the k-th canonical basis of $\mathbb{C}^n$. Therefore, the $\mathcal{P}_\Omega$ is a projection operator. The sampling rate is then $Sp = m/n \leq 1$.

2.1.3 The Spectrally Sparse Signal Model

Many applications involve smooth or modulated signals that can be represented as a combination of r sinusoidal components.We consider an order-r SSS $\boldsymbol{x} \in \mathbb{C}^n$ (with $r \ll n$), representing a

superposition of complex sinusoids:

$$\boldsymbol{x} = \sum_{k=1}^{r} d_k \boldsymbol{y}(f_k, \tau_k;\ n), \tag{2.2}$$

where $d_k \in \mathbb{C}$ is the complex amplitude, and $\boldsymbol{y}(f_k, \tau_k;\ n) \coloneqq [1, e^{i2\pi f_k - \tau_k}, \ldots, e^{(i2\pi f_k - \tau_k)(n-1)}]^{\mathsf{T}} \in \mathbb{C}^n$ represents the kth spectral component with normalized frequency $f_k \in [0, 1)$ and damping factor τ_k [56]. The SSS model is a widely used model in the field of *super-resolution*. It is a generalization version of the sparse signal model, which is a fundamental concept in CS. The sparse signal model is based on the premise that the signal of interest can be represented as a linear combination of a few atoms from a dictionary. The SSS model extends this concept by considering a signal that is sparse across a continuous spectrum. The applicability of this model is broad, spanning various domains such as speech processing, power system monitoring, and computing imaging. Additionally, it plays a crucial role in the estimation of direction-of-arrival in uniform/sparse linear arrays, an area that has received considerable attention in the spectrum analysis literature [27].

Remark 2. *In the case of multidimensional SSS, represented as* $\boldsymbol{X} = \sum_k d_k \boldsymbol{y}(f_{1,k}, \tau_{1,k};\ n_1) \circ \cdots \circ \boldsymbol{y}(f_{d,k}, \tau_{d,k};\ n_d) \in \mathbb{C}^{n_1 \times \cdots \times n_d}$ *where* $\circ$ *denotes the outer product, a multilevel Hankel matrix is required.*

Another Model of Super-resolution, Equivalent to the Line Spectrum Estimation

To be more precise, we initiate our discussion by delving into a fundamental and widely applicable *super-resolution* model as introduced in [1]. This model encapsulates the convolution of a sequence of point sources PSF that is limited in resolution, as illustrated in Figure 2.1. Consider the formulation of a spike signal $x(t)$, which is defined as follows:

$$x(t) = \sum_{k=1}^{r} d_k \hat{\delta}(t - f_k). \tag{2.3}$$

In this expression, r signifies the total number of spikes. The terms $d_k \in \mathbb{C}$ and f_k[1] $\in [0, 1)$ represent the complex amplitude and the delay of the k-th spike, respectively. The symbol $\hat{\delta}$, also known as the unit impulse, is a generalized function or distribution across the real numbers. It is characterized by being zero everywhere except at zero and having its integral over the entire real line equal to one. This model of a spike signal is versatile and can represent a variety of physical phenomena, ranging from the timing of neuronal firings to the localization of fluorescence molecules.

The PSF, denoted as $g(t)$, is subject to bandwidth limitations imposed by the Rayleigh Limit. The Fourier transform of the PSF, represented as $G(f)$, satisfies the condition $G(f) = 0$ for $|f| > \frac{B}{2}$, where B is the defined bandwidth and is greater than zero. The convolution of the spike signal $x(t)$ with the PSF $g(t)$, which is further affected by additive noise $\varepsilon(t)$, can be mathematically expressed as follows:

$$y(t) = x(t) * g(t) + \varepsilon(t) = \sum_{k=1}^{r} d_k g(t - f_k) + \varepsilon(t), \tag{2.4}$$

where $*$ denotes the convolution operation. This equation characterizes the resultant signal $y(t)$ as a combination of the convolved spike signal with the PSF and the noise component $\varepsilon(t)$.

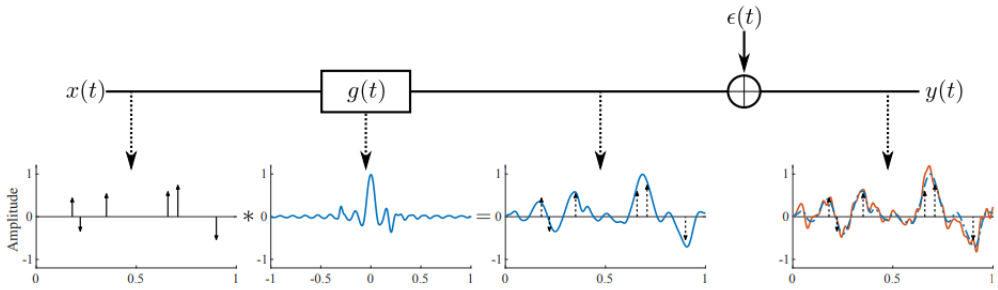


Figure 2.1: An illustration of the mathematical model of super-resolution. The spike signal $x(t)$ is convolved with a PSF $g(t)$, leading to degradation of its resolution, which is further exacerbated by an additive noise $\varepsilon(t)$, producing an output signal $y(t)$ [1].

By sampling the Fourier transform of the aforementioned equation at specific frequencies, denoted as $\ell = -\lfloor \frac{B}{2} \rfloor, \ldots, 0, \ldots, \lfloor \frac{B}{2} \rfloor$, we can obtain the corresponding measurements. These

[1]We use f_k here to match the notation in the SSS model.

measurements are mathematically expressed as:

$$Y_\ell = G_\ell \cdot X_\ell + E_\ell = G_\ell \cdot \left(\sum_{k=1}^{r} d_k e^{-i2\pi\ell f_k} \right) + E_\ell, \tag{2.5}$$

where Y_ℓ represents the measured signal at frequency ℓ. In this equation, G_ℓ and X_ℓ denote the Fourier transforms of the PSF $g(t)$ and the spike signal $x(t)$ at frequency ℓ, respectively, while E_ℓ symbolizes the noise component in the frequency domain. The expression effectively captures the relationship between the observed signal, the PSF, the spike signal, and the noise in the frequency domain. The total count of samples captured in this model is denoted as $n = 2\lfloor \frac{B}{2} \rfloor + 1$, which approximately equals B. This approximation helps in simplifying the analysis. Subsequently, Equation (2.5) can be reformulated in a vectorized format as follows:

$$\boldsymbol{y} = \mathrm{diag}(\boldsymbol{g})\boldsymbol{x} + \boldsymbol{\varepsilon}, \tag{2.6}$$

where $\boldsymbol{y} = [Y_\ell]$ represents the vector of measured signals, $\boldsymbol{g} = [G_\ell]$ corresponds to the Fourier transforms of the PSF, $\boldsymbol{x} = [X_\ell]$ signifies the spike signal in the frequency domain, and $\boldsymbol{\varepsilon} = [E_\ell]$ is the noise vector. A crucial aspect of the *super-resolution* problem is the accurate estimation of the set $\{d_k, f_k\}_{k=1}^{r}$ from the observed data $\boldsymbol{y}$, without prior knowledge of the number of spikes, r. The Rayleigh limit, inversely proportional to the bandwidth B, is approximately equivalent to $1/n$, establishing a relationship between the resolution limit and the number of samples.

When the PSF $g(t)$ is known, Equation (2.5) can undergo "equalization" by multiplying both sides with the inverse of G_ℓ, provided that G_ℓ values are nonzero. This operation yields a transformed observation $\boldsymbol{s} = [G_\ell^{-1} Y_\ell]$, which can be related to the spike signal $\boldsymbol{x}$ as follows:

$$\boldsymbol{s} = \boldsymbol{x} + \boldsymbol{\varepsilon}_e, \tag{2.7}$$

where $\boldsymbol{\varepsilon}_e$ denotes the transformed additive noise.

For analytical convenience, we remap the index ℓ from $-\lfloor B/2 \rfloor, \ldots, \lfloor B/2 \rfloor$ to $0, \ldots, n-1$. The

spike signal $\boldsymbol{x}$ is then expressed as a summation of complex sinusoids:

$$\boldsymbol{x} = \sum_{k=1}^{r} d_k \boldsymbol{a}(f_k), \tag{2.8}$$

where $\boldsymbol{a}(f) \in \mathbb{C}^n$ is defined as:

$$\boldsymbol{a}(f) = \left[1, e^{i2\pi f}, \ldots, e^{i2\pi(n-1)f}\right]^{\mathsf{T}}, \quad f \in [0, 1). \tag{2.9}$$

This vector $\boldsymbol{a}(f) = \boldsymbol{y}(f_k, 0;\ n)$ encapsulates the frequency components of the signal, offering a compact and efficient representation for the analysis in the frequency domain.

Remarkably, the simplified model depicted in Equation (2.7) aligns with the classical problem of line spectrum estimation. This estimation problem involves deducing a mixture of sinusoids, characterized by frequencies $f_k \in [0, 1)$, from equi-spaced time samples $\{0, \ldots, n-1\}$ of the signal $x_{ls}(t) = \sum_{k=1}^{r} d_k e^{i2\pi f_k t}$. This model's versatility and foundational nature underscore its importance in the field of signal processing.

2.2 Atomic Norm Minimization

In the simplest case, where noise is absent, the concept of *super-resolution* can be framed as the task of inferring the continuous-time spike signal, $x(t)$, as defined in Equation (2.3), from its discrete-time moment measurements, denoted as $\boldsymbol{x}$ in Equation (2.8). The relationship between these measurements can be represented as follows:

$$\boldsymbol{x} = \int_0^1 \boldsymbol{a}(f)\, dx(t) \tag{2.10}$$

In this context, $x(t)$ is perceived as a continuous spectrum representation of $\boldsymbol{x}$, outlined by the dictionary $\mathcal{A}_0$ as specified in Equation 2.11:

$$\mathcal{A}_0 = \{\boldsymbol{a}(f) : f \in [0, 1)\}. \tag{2.11}$$

The moment curve described here forms a one-dimensional manifold within $\mathbf{C}^n$. It is well-acknowledged that the convex hull of $\mathcal{A}_0$ within $\mathbf{C}^n$ can be characterized using a series of linear matrix inequalities, as detailed in [69]. This characterization is notably linked to the concept of positivity in Hermitian Toeplitz matrices. The inherent properties of the moment curve are fundamental in various domains, including control and signal processing [70, 71]. This aspect is instrumental in developing a *super-resolution* theory based on ANM, providing a robust theoretical foundation for resolving signals at a resolution beyond conventional limits.

While different instances of $x(t)$ can yield the same measurement vector $\boldsymbol{x}$, enforcing a sparsity constraint—specifically, limiting the number of spikes in $x(t)$—facilitates a unique representation. Notably, the representation in (2.8) is unique if $r \leq \frac{\lfloor n \rfloor}{2}$ and the support set $\mathcal{F} = \{f_k\}_{k=1}^r$ contains distinct elements.

In order to apply the ANM framework effectively to *super-resolution* problems, it is crucial to define the atomic set accurately. Given that the complex amplitudes d_k can vary in phase, we introduce an atomic set to encompass this variability:

$$\mathcal{A} = \{\boldsymbol{a}(f, \phi) : f \in [0, 1), \phi \in [0, 2\pi)\}, \tag{2.12}$$

where $\boldsymbol{a}(f, \phi) = e^{i\phi}\left[1, e^{i2\pi f}, \ldots, e^{i2\pi(n-1)f}\right]^\mathsf{T}$, Here, the set of atoms serves as the fundamental building blocks for the signal $\boldsymbol{x}$, akin to how canonical basis vectors are for sparse signals and unit-norm rank one matrices are for low-rank matrices. In sparse recovery and matrix completion, the unit balls of sparsity-enforcing norms, such as the ℓ_1 norm and nuclear norm, correspond exactly to the convex hulls of their respective building blocks. Similarly, we define an atomic norm $\|\cdot\|_{\mathcal{A}}$ by associating its unit ball with the convex hull of $\mathcal{A}$:

$$\begin{aligned} \|\boldsymbol{x}\|_{\mathcal{A}} &\coloneqq \inf\{t > 0 : \boldsymbol{x} \in t \cdot \operatorname{conv}(\mathcal{A})\} \\ &= \inf_{d_k \geq 0, \phi_k \in [0,2\pi), f_k \in [0,1)} \left\{ \sum_k d_k : \boldsymbol{x} = \sum_k d_k \boldsymbol{a}(f_k, \phi_k) \right\}. \end{aligned} \tag{2.13}$$

Essentially, the atomic norm $\|\cdot\|_{\mathcal{A}}$ promotes sparsity in $\mathcal{A}$ because signals involving fewer atoms correspond to low-dimensional facets of $\operatorname{conv}(\mathcal{A})$. This concept of leveraging atomic norms for

general sparsity enforcement was first proposed and explored in [36]. As $\mathcal{A}$ is centrally symmetric around the origin, it induces an atomic norm over $\mathbf{C}^n$, as outlined in equations (2.13).

Interestingly, minimizing the atomic norm of $\boldsymbol{x}$ aligns with minimizing the total variation of $x(t)$ [33], represented as:

$$\min \|x(t)\|_{\mathrm{TV}} \quad \text{s.t.} \quad \boldsymbol{x} = \int_0^1 \boldsymbol{a}(t)\, dx(t). \tag{2.14}$$

Both the atomic norm and total variation perspectives are widely adopted in contemporary literature.

Remarkably, the atomic norm admits an equivalent semidefinite program (SDP) characterization. This equivalence arises due to the Carathéodory-Fejér-Pisarenko decomposition [72] and its extension to include phases [35]:

$$\|\boldsymbol{x}\|_{\mathcal{A}} = \inf_{\boldsymbol{u}\in\mathbb{C}^n} \left\{ \frac{1}{2n}\mathrm{Tr}(\mathrm{toep}(\boldsymbol{u})) + \frac{1}{2}t : \begin{bmatrix} \mathrm{toep}(\boldsymbol{u}) & \boldsymbol{x} \\ \boldsymbol{x}^H & t \end{bmatrix} \succeq 0, t > 0, \boldsymbol{u} \in \mathbb{C}^n \right\}, \tag{2.15}$$

where $\mathrm{toep}(\boldsymbol{u})$ represents the Hermitian Toeplitz matrix[2] with its first column being $\boldsymbol{u}$.

In contrast to its abstract form in (2.13), the reformulation in (2.15) offers a tractable method to compute the atomic norm $\|\boldsymbol{x}\|_{\mathcal{A}}$, achievable using standard convex solvers [73]. Employing the Vandermonde decomposition of $\mathrm{toep}(\boldsymbol{u})$, i.e., $\mathrm{toep}(\boldsymbol{u}) = \sum_{k=1}^{r'} |d'_k| a(f'_k) a(f'_k)^H$, allows for the identification of the support $\hat{\mathcal{F}} = \{f'_k\}$ of $\boldsymbol{x}$'s atomic representation, as well as the atomic norm $\|\boldsymbol{x}\|_{\mathcal{A}} = \sum_{k=1}^{r'} |d'_k|$. The computation of the Vandermonde decomposition of the Toeplitz matrix can be efficiently performed via the Lanczos method [74], requiring $\mathcal{O}(n^2)$ operations.

Moreover, the Lagrange dual problem associated with the ANM in (2.15) offers several advantages:

1. The dual problem can also be reformulated into an SDP;

[2] A Toeplitz matrix is characterized by constant descending diagonals from left to right.

2. The support set of the atomic decomposition can be deduced from the solution of the dual problem.

In this discourse, we have examined the application of ANM in the *super-resolution* of line spectra. Central to our discussion is the critical issue of the tightness of the convex relaxation method employed in ANM. Specifically, our objective is to identify the precise conditions under which the estimated support $\hat{\mathcal{F}}$ accurately coincides with the true support $\mathcal{F}$ of the signal $\boldsymbol{x}$. Such alignment is imperative not only for the support sets but also for the atomic decompositions. In other words, we aim to ensure that the representation $\boldsymbol{x} = \sum_{k=1}^{r'} d'_k a(\tau'_k)$ aligns with the most sparse form $\boldsymbol{x} = \sum_{k=1}^{r} d_k a(\tau_k)$ within the atomic set $\mathcal{A}$.

This aspect is vital as it underpins the effectiveness of ANM in yielding a representation that is not only accurate but also the sparsest possible within the given atomic framework. The clarification of such conditions would mark a significant advance in our understanding and application of *super-resolution* techniques, particularly in contexts where precision and sparsity are of paramount importance.

2.2.1 Exact Recovery Guarantees

The investigation of conditions conducive to exact recovery has been extensive, particularly in the realm of ℓ_1 norm minimization, which is relevant when the atomic set $\mathcal{A}$ is finite. In such scenarios, performance guarantees are often linked to the structural characteristics of the set. These characteristics are typically defined in terms of concepts such as the Restricted Isometry Property (RIP) [75] or distinct incoherence properties [76]. However, challenges arise when dealing with a continuous dictionary, such as $\mathcal{A}$. In these cases, the effectiveness of these properties diminishes. This reduction in efficacy is due to the increasing correlation between atoms $a(\tau)$ and $a(\tau + \delta)$ as their separation δ becomes smaller, resulting in significantly reduced RIP or coherence levels.

A significant finding in this area [33] is the possibility of deterministically establishing a valid

certificate for sufficiently large n, depending upon the fulfillment of the separation condition. This finding is notable for its independence from the spikes' complex amplitudes and does not necessitate relying on any stochastic characteristics of the signal. It suggests that under certain well-defined conditions, exact recovery is feasible, thereby providing a robust foundation for the application of atomic norm minimization in *super-resolution* tasks.

2.2.2 The Noisy Cases

In practical scenarios, the presence of noise inevitably distorts observational data, rendering it impossible for any estimator to perfectly reconstruct the spike signal $x(t)$. This reality necessitates a careful consideration of the robustness of estimates produced by atomic norm minimization techniques in noisy environments. When faced with additive noise in the observed signal $\boldsymbol{s}$, as characterized in Equation (2.7), a viable strategy involves seeking an $\boldsymbol{x}$ that lies close to $\boldsymbol{s}$ while having a minimal atomic norm [37]:

$$\min_{\boldsymbol{x}} \frac{1}{2}\|\boldsymbol{x}-\boldsymbol{s}\|_2^2 + \lambda\|\boldsymbol{x}\|_{\mathcal{A}}, \tag{2.16}$$

where $\lambda > 0$ is a parameter that strikes a balance between fidelity to the observed data and the magnitude of the atomic norm. The choice of this parameter is less clear [37, 40, 77]. This method, known as "atomic norm denoising", is an adaptation of the principles underlying the well-established LASSO technique [78].

Despite the computational feasibility of the SDP framework, its complexity can become quite demanding in high-dimensional settings. To mitigate this, the Alternating Direction Method of Multipliers (ADMM) [79], as employed in [1], offers significant improvements in computational speed. ADMM functions by breaking down the augmented Lagrangian of an optimization problem into separate, manageable subproblems. Each iteration of the algorithm involves localized minimization of these subproblems, while maintaining compliance with the overall constraints. The iterative process continues until the discrepancies in both the primal and dual solutions are reduced below a set threshold. Typically, the computation is executed with a

complexity of $\mathcal{O}(n^3)$ per iteration, rendering it a more efficient alternative for high-dimensional problems.

2.2.3 Spectral Compressed Sensing

The ANM is also applicable to the SCS problem, which is a generalization of the frequency estimation problem. The SCS problem is defined as follows: given a set of indices $\boldsymbol{\Omega} = \{\omega_1, \ldots, \omega_m\}$ for the known entries of $\boldsymbol{x}$, the challenge is to reconstruct $\boldsymbol{x}$ from $\boldsymbol{s} \coloneqq \mathcal{P}_\Omega \boldsymbol{x} \in \mathbb{C}^n$. The $\mathcal{P}_\Omega \boldsymbol{x}$ is the projection of $\boldsymbol{x}$ onto the subspace of arrays that vanish outside Ω. It can be formulated as a convex optimization problem:

$$\min_{\boldsymbol{x}} \|\boldsymbol{x}\|_{\mathcal{A}} \quad \text{s.t.} \quad \mathcal{P}_\Omega \boldsymbol{x} = \boldsymbol{s}, \tag{2.17}$$

Specifically, when only m entries of $\boldsymbol{x} \in \mathbb{C}^n$ are observed uniformly at random, there exists a high probability of perfectly recovering $\boldsymbol{x}$. This recovery is feasible using $m = \mathcal{O}(\log^2 n + r \log r \log n)$ measurements, provided $\boldsymbol{x}$ adheres to the separation condition and the coefficients d_k's have random signs [35][Theorem II.3].

2.3 Multi-block Hankel Matrices Involved Algorithms

While the atomic norm provides a viable approach to exploit sparsity in continuous domains, it has several drawbacks that limit its applicability in certain scenarios:

1. The atomic norm approach fails to accommodate the damping factor in the signal model, a feature crucial in numerous applications such as NMR [80]. Modeling NMR data as multidimensional damped sinusoids is both a natural reflection of the physical behavior of nuclear spins in a magnetic field and a practical approach to analytically describing and analyzing the complex decaying signals generated in different types of NMR experiments.

2. The matrix involved in atomic norm minimization is not a specific structured form (Hankel

or Toeplitz) although partially included, which can not apply efficient computational strategies. In cases of large data sizes, the unstructured matrix required for atomic norm minimization becomes challenging to manage.

3. Extending the atomic norm minimization framework to multi-dimensional cases is not straightforward, as indicated in [1, 2]. However,the problem of estimating multidimensional frequencies emerges in multiple applications such as array processing, radar, sonar, astronomy, and medical imaging [39]. For example in the seismic data processing, a level-4 Toeplitz matrix is formed which depends on four spatial dimensions [81].

Fortunately, the recent advancements in Enhanced Matrix Completion (EMaC) [2] offer a promising alternative. EMaC leverages the multi-fold Hankel structure to exploit both the shift-invariance property of harmonic structures and the spectral sparsity of signals. By organizing data samples into a multi-block Hankel matrix, the rank of this matrix is effectively constrained by the signal's spectral sparsity. This approach not only addresses the limitations of the atomic norm minimization in handling damping factors and large-scale data but also provides a more straightforward pathway to extending the methodology to multi-dimensional signal processing scenarios.

To facilitate understanding, let us consider the two-dimensional frequency models as an example to elucidate the concept of the multi-block Hankel matrix and the related algorithms. Imaging a signal $\boldsymbol{X}(t)$ that is composed of *two*-dimensional complex sinusoids. This signal incorporates r distinct frequencies $f_k \in [0,1)^2$ and corresponding damping factors τ_k for each $1 \leq k \leq r$. Consequently, a two-dimensional spectrally sparse array $\boldsymbol{x} \in \mathbb{C}^{n_1 \times n_2}$ can be formulated as:

$$\boldsymbol{X} = \sum_{k=1}^{r} d_k \boldsymbol{y}(f_{1k}, \tau_{1k};\ n_1) \circ \boldsymbol{y}(f_{2k}, \tau_{2k};\ n_2) \in \mathbb{C}^{n_1 \times n_2}, \tag{2.18}$$

where $d_k \in \mathbb{C}$ are complex amplitudes, and $\boldsymbol{y}(f_k, \tau_k;\ n) \coloneqq [1, e^{i2\pi f_k - \tau_k}, \ldots, e^{(i2\pi f_k - \tau_k)(n-1)}]^{\mathsf{T}} \in \mathbb{C}^n$ $(n_1 + n_2 = n + 1)$. The array $\boldsymbol{F}_k = (f_{1k}, f_{2k}) \in [0,1)^2$ denotes the normalized frequencies with respect to the Nyquist frequency of $\boldsymbol{X}(t)$, and $\tau_{1k}, \tau_{2k} \geq 0$ are the damping factors, respectively. Given a set of indices $\boldsymbol{\Omega} = \{(\omega_1, \omega_2) \in [n_1] \times [n_2]\}$ that correspond to the known entries of $\boldsymbol{x}$,

the challenge lies in reconstructing $\boldsymbol{X}$ from its partial observations $\boldsymbol{S} = \mathcal{P}_\Omega \boldsymbol{X} \in \mathbb{C}^{n_1 \times n_2}$.

This two-dimensional framework extends the traditional SCS approach to accommodate complex, multi-dimensional signals and their sparse representations, thereby enabling more efficient and accurate reconstruction in various applications.

2.3.1 2-D Vandermonde Decomposition

Each element $\boldsymbol{x}_{m,n}$ of the matrix $\boldsymbol{X}$ $(0 \leq m < n_1, 0 \leq n < n_2)$ can be represented as follows:

$$\boldsymbol{x}_{m,n} = \boldsymbol{X}(m,n) = \sum_{k=1}^{r} d_k y_{1k}^m y_{2k}^n, \tag{2.19}$$

where, for each index k (ranging from 1 to r):

$$y_{1k} \coloneqq \exp(i2\pi f_{1k} - \tau_{1k}) \quad \text{and} \quad y_{2k} \coloneqq \exp(i2\pi f_{2k} - \tau_{2k}), \tag{2.20}$$

correspond to the pairs of frequencies $\{f_k = (f_{1k}, f_{2k}) | 1 \leq k \leq r\}$. Consequently, the matrix $\boldsymbol{X}$ can be decomposed as

$$\boldsymbol{X} = \boldsymbol{Y}_1 \boldsymbol{D} \boldsymbol{Y}_2^{\mathsf{T}}, \tag{2.21}$$

where the matrices $\boldsymbol{Y}_1$, $\boldsymbol{Y}_2$, and $\boldsymbol{D}$ are defined respectively as:

$$\boldsymbol{Y}_1 = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ y_1 & y_2 & \cdots & y_r \\ \vdots & \vdots & \ddots & \vdots \\ y_1^{n_1-1} & y_2^{n_1-1} & \cdots & y_r^{n_1-1} \end{bmatrix}, \quad \boldsymbol{Y}_2 = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ y_1 & y_2 & \cdots & y_r \\ \vdots & \vdots & \ddots & \vdots \\ y_1^{n_2-1} & y_2^{n_2-1} & \cdots & y_r^{n_2-1} \end{bmatrix},$$

and

$$\boldsymbol{D} = \operatorname{diag}[d_1, d_2, \cdots, d_r].$$

This decomposition of $\boldsymbol{X}$, as expressed in Equation (2.21), is often referred to as the Vandermonde decomposition. It provides a structured and compact representation of the 2-D signal, facilitating

efficient computational approaches for signal reconstruction and analysis.

Inspired by the advancements in matrix completion (MC) [50], it is natural to consider the application of low-rank MC algorithms, particularly when the rank r is small. The potential for perfect recovery of $\boldsymbol{X}$ from partial measurements becomes viable when $\boldsymbol{X}$ exhibits a low rank, specifically if $r \ll \min\{n_1, n_2\}$. This consideration leads to the adoption of the following optimization framework:

$$\text{minimize} \quad \|\boldsymbol{S}\|_* \tag{2.22}$$

$$\text{subject to} \quad P_\Omega \boldsymbol{S} = P_\Omega \boldsymbol{X}, \tag{2.23}$$

where $\|\boldsymbol{S}\|_*$ denotes the nuclear norm (the sum of all singular values) of the matrix $\boldsymbol{S}$. This strategy forms a convex relaxation alternative to the direct minimization of rank. However, traditional MC algorithms, as described in [82], typically require at least $r \max\{n_1, n_2\} \log(n_1 n_2)$ samples to achieve perfect recovery. This sample size surpasses the inherent degrees of freedom, $\mathcal{O}(r)$, in our problem setup. A critical observation is that as the number r of spectral spikes nears $n_1 n_2$, the matrix X may approach a full-rank status, particularly when $r > \min\{n_1, n_2\}$. This scenario underscores the need to explore alternative representations that more effectively capture the harmonic structure inherent in the problem.

2.3.2 Enhanced Hankel Matrix

In response to these challenges, the Enhanced Matrix Completion (EMaC) strategy [2] has been proposed to effectively address the problem. The EMaC approach constructs an enhanced two-fold Hankel matrix for $\boldsymbol{X}$ through a recursive process, as described below:

$$\boldsymbol{X}^e = \mathcal{H}\boldsymbol{X} = \begin{bmatrix} \mathcal{H}\boldsymbol{X}[:,0] & \mathcal{H}(\boldsymbol{X}[:,1]) & \dots & \mathcal{H}(\boldsymbol{X}[:,q_2-1]) \\ \mathcal{H}(\boldsymbol{X}[:,1]) & \mathcal{H}(\boldsymbol{X}[:,2]) & \dots & \mathcal{H}(\boldsymbol{X}[:,q_2]) \\ \vdots & \vdots & \ddots & \vdots \\ \mathcal{H}(\boldsymbol{X}[:,p_2-1]) & \mathcal{H}(\boldsymbol{X}[:,p_2]) & \dots & \mathcal{H}(\boldsymbol{X}[:,n_2-1]) \end{bmatrix} \in \mathbb{C}^{p_1 p_2 \times q_1 q_2},$$

where $\mathcal{H}$ denotes the Hankel operator, $p_2\,(1 \leq p_2 \leq n_2)$ is the pencil parameter, and $\boldsymbol{X}[:n]$, $1 \leq n \leq n_2$, represents the n-th slice of $\boldsymbol{X}$. Each element of this matrix is defined as

$$\mathcal{H}(\boldsymbol{X}[:,n]) = \begin{bmatrix} \mathcal{H}(\boldsymbol{X}[0,n]) & \dots & \mathcal{H}(\boldsymbol{X}[q_1-1,n]) \\ \vdots & \ddots & \vdots \\ \mathcal{H}(\boldsymbol{X}[p_1-1,n]) & \dots & \mathcal{H}(\boldsymbol{X}[n_1-1,n]) \end{bmatrix},$$

with $p_1 + q_1 = n_1 + 1$, $p_2 + q_2 = n_2 + 1$, and $\mathcal{H}(\boldsymbol{X}[:,n]) \in \mathbb{C}^{p_1 \times q_1}$. Here, $p_1(1 \leq p_1 \leq n_1)$ is another pencil parameter.

This EMaC approach, through its unique construction of the enhanced Hankel matrix, offers a more structured and potentially more computationally efficient way to harness the inherent spectral sparsity and harmonic structure of the signal $\boldsymbol{X}$.

An explicit formula for the enhanced Hankel matrix $\mathcal{H}\boldsymbol{X}$ is defined as follows:

$$[\mathcal{H}\boldsymbol{X}]_{uv} = \boldsymbol{X}[m,n],$$

where the indices are determined by the equations:

$$\begin{aligned} u &= u_1 + u_2 \cdot p_1, \quad & v &= v_1 + v_2 \cdot (n_1 - p_1 + 1), \\ m &= u_1 + v_1, & n &= u_2 + v_2. \end{aligned}$$

Furthermore, $\mathcal{H}\boldsymbol{X}$ possesses a Vandermonde decomposition of the form $\mathcal{H}\boldsymbol{X} = \boldsymbol{E_L} \boldsymbol{D} \boldsymbol{E_R}^{\mathsf{T}}$. The k-th columns of $\boldsymbol{E_L}$ and $\boldsymbol{E_R}$ are given by:

$$\begin{aligned} \boldsymbol{E_L}[:,k] &= \{\boldsymbol{y}(f_{1k}, \tau_{1k};\ n_1)[m] \cdot \boldsymbol{y}(f_{2k}, \tau_{2k};\ n_2)[n] \mid 1 \leq m \leq p_1, 1 \leq n \leq p_2\}, \\ \boldsymbol{E_R}[:,k] &= \{\boldsymbol{y}(f_{1k}, \tau_{1k};\ n_1)[m] \cdot \boldsymbol{y}(f_{2k}, \tau_{2k};\ n_2)[n] \mid 1 \leq m \leq q_1, 1 \leq n \leq q_2\}, \end{aligned}$$

and $\boldsymbol{D} = \mathrm{diag}(d_1, \cdots, d_r)$. As a result, $\mathcal{H}\boldsymbol{X}$ remains a rank r matrix for two-dimensional arrays.

The problem of recovering $\boldsymbol{X}$ in EMaC from partial measurements is then formulated as:

$$\text{minimize} \quad \|\boldsymbol{M}\|_* \tag{2.24}$$

$$\text{subject to} \quad P_\Omega^e \boldsymbol{M} = P_\Omega^e \boldsymbol{X}^e, \tag{2.25}$$

where P_Ω^e is the corresponding sampling operator on the enhanced matrix space. This problem aims to recover the low-rank matrix $\boldsymbol{X}^e$ from partial measurements by minimizing the nuclear norm. This convex optimization can be transformed into a semidefinite programming (SDP) problem and solved using off-the-shelf solvers [83].

2.3.3 Singular Value Thresholding for EMaC

While the computational complexity of the Enhanced Matrix Completion (EMaC) method is similar to that of the ANM approach [35], particularly for one-dimensional frequency models, the standard SDP solvers, which often employ interior point methods, face difficulties with large-scale data. For instance, SDPT3 struggles with an $n \times n$ data matrix when n exceeds 19, which corresponds to a 100×100 matrix post-enhancement. An effective solution for handling large-scale data is to apply first-order methods specifically tailored for matrix completion challenges, such as the Singular Value Thresholding (SVT) algorithm [84].

The SVT algorithm is a first-order method designed to solve the following convex optimization problem efficiently:

Algorithm 1 Singular Value Thresholding for EMaC [2]

Input: The observed data matrix $\boldsymbol{x}_0$ on the location set Ω;
Initialization: Initialize the enhanced matrix $\boldsymbol{X}_0^e = \mathcal{H}\boldsymbol{x}_0$; set $\boldsymbol{M}_0 = \boldsymbol{X}_0^e$ and $t = 0$;
  while not converged **do**
    $\boldsymbol{Q}_t \leftarrow \mathcal{T}_{\tau_t} \boldsymbol{M}_t$.
    $\boldsymbol{M}_t \leftarrow \mathcal{H}(\mathcal{P}_\Omega \mathcal{H}^\dagger \boldsymbol{Q}_t + \boldsymbol{x}_0)$.
  end while
Output: Retrieve $\boldsymbol{M}_t$ from the final iteration, set $\boldsymbol{x}^\dagger \leftarrow \mathcal{H}^\dagger \boldsymbol{M}_t$.

The thresholding operator $\mathcal{T}_{\tau_t}(\cdot)$ in Algorithm 1 denotes the singular value shrinkage operator. Given the Singular Value Decomposition (SVD) of a matrix $\boldsymbol{X}^e = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^*$, with $\boldsymbol{\Sigma} = \text{diag}(\{\sigma_i\})$,

then $\mathcal{T}_{\tau_t}\boldsymbol{X}^e$ is defined as $\boldsymbol{U}\text{diag}(\{\max(\sigma_i - \tau_t, 0)\})\boldsymbol{V}^*$, where $\tau_t > 0$ is a soft-thresholding parameter.

Remark 3. *In each iteration, the algorithm constructs a pair $\boldsymbol{Q}_t, \boldsymbol{M}_t$ by first applying singular value shrinkage and then projecting the result onto the space of multi-level Hankel matrices that align with the observed entries. Importantly, all matrices involved in the algorithm possess a multilevel Hankel structure, facilitating efficient storage and computation. Consequently, the computational complexity of EMaC is reduced from $\mathcal{O}(n^3)$ to $\mathcal{O}(n^2\log(n))$ compared to ANM, making it a more feasible option for large-scale data sets.*

2.3.4 Advantages of Enhanced Hankel Matrix Approaches

In the field of *super-resolution* compressed sensing, two principal methods have been predominantly reviewed: the ANM [35] and the EMaC [2]. To gain a deeper insight into EMaC, we explore structured low-rank matrix representations that promote spectral sparsity from the perspective of atomic norms. This exploration involves formulating these methods as atomic norm minimizations, which arise from various choices of atomic sets in Cadzow's signal enhancement framework. This viewpoint enables us to interpret EMaC, also known as structured Hankel matrix completion, as a form of relaxed atomic norm minimization tailored for complex sinusoids.

Consider an order-r spectrally sparse signal (SSS) defined in the Eq. 2.2.

Define the atomic set $\mathcal{S}_0$ as:

$$\mathcal{S}_0 = \left\{ \sum_{k=1}^{r} d_k \boldsymbol{y}(f_k, 0;\ n) : f_k \in [0,1), d_k \in \mathbb{C} \right\}. \tag{2.26}$$

Owing to physical or measurement constraints, often only a subset of $\boldsymbol{x}$ is observable. The primary challenge lies in recovering the SSS $\boldsymbol{x}$ from its partial observations $\boldsymbol{s} \in \mathbb{C}^n$, formulated as:

$$\text{find} \quad \boldsymbol{x} \in \mathcal{S}_0 \quad \text{s.t.} \quad \mathcal{P}_\Omega \boldsymbol{x} = \boldsymbol{s}. \tag{2.27}$$

As elucidated in [35, 39], the set $\mathcal{S}_0$ is equivalently represented by the Toeplitz-related matrix set $\mathcal{S}_T$:

$$\mathcal{S}_T = \left\{ \boldsymbol{x} \in \mathbb{C}^n : \begin{bmatrix} \text{toep}(\boldsymbol{u}) & \boldsymbol{x} \\ \boldsymbol{x}^H & t \end{bmatrix} \succeq 0, \boldsymbol{u} \in \mathbb{C}^n, \text{rank}(\text{toep}(\boldsymbol{u})) \leq r, t > 0 \right\}. \tag{2.28}$$

Here, ANM can be formulated as a convex optimization problem for this PSD Hermitian Toeplitz (toep($\boldsymbol{u}$)) model.

Similarly, the low-rank Hankel model can be interpreted as the quest for the best approximation within the Hankel-based set:

$$\mathcal{S}_H = \{ \boldsymbol{x} \in \mathbb{C}^n : \text{rank}(\mathcal{H}\boldsymbol{x}) \leq r \}. \tag{2.29}$$

In this context, EMaC represents a convex relaxation in accordance with $\mathcal{S}_H$.

Given the inclusion $\mathcal{S}_T \subseteq \mathcal{S}_H$, the relationship between ANM and EMaC is highlighted in the following proposition [46, Prop. 2.1]: *The nuclear norm minimization, denoted by $\|\mathcal{H}\boldsymbol{x}\|_*$, acts as a convex surrogate for the atomic norm minimization on $\mathcal{S}_T$.*

This proposition emphasizes the interconnectedness of ANM and EMaC, demonstrating how the latter can be seen as a convex relaxation of the former, particularly within the framework of structured low-rank matrix representations that aim to promote spectral sparsity.

The relaxation to a larger set, $\mathcal{S}_H$, presents several advantages, as outlined below:

1. The EMaC approach demonstrates superior resolution compared to the ANM method, as substantiated in [2, 85]. This improvement is attributed to a significantly milder coherence condition.

2. The Hankel-based model permits the application of various non-convex methods [56–58, 61, 86]. These approaches are typically more computational efficiently than their convex counterparts.

3. The broader scope of $\mathcal{S}_H$ makes it applicable to a wider array of problems, including NMR signals with damping factors [15] and other Hankel matrix-involved problems.

Remark 4. *The studies in [43, 85] also explore the Hankel-Toeplitz model and the double Hankel model. These models are formulated to capture spectral sparsity and can be directly solved. However, their applicability is primarily limited to undamped signals. In the future, it may be possible to extend PG-based methods to these models, broadening their scope and utility.*

2.3.5 Linear Recurrent Time Series

As previously demonstrated, the low-rank structure of Hankel matrices is pivotal in the development of the EMaC method. It is not limited to spectrally sparse signal models but also extends to other low-rank Hankel completion problems including time series represented as $\boldsymbol{x} = [x_0, \ldots, x_{n-1}]^{\mathsf{T}}$ with length n, which conform to a linear recurrent formula (LRF) if:

$$x_j = a_1 x_{j-1} + \cdots + a_r x_{j-r}, \quad j = r, \ldots, n-1. \tag{2.30}$$

Let $p < n$ be an integer, typically denoted as the window length, and $q = n - p + 1$. Define the p-lagged vectors $\boldsymbol{x}_j = [x_j, \ldots, x_{j+p-1}]^{\mathsf{T}}$, for $j = 0, \ldots, q-1$. By constructing a $p \times q$ matrix $\boldsymbol{X}^e$ as

$$\boldsymbol{X}^e = [\boldsymbol{x}_0, \ldots, \boldsymbol{x}_{q-1}],$$

it becomes evident that $\boldsymbol{X}^e$ is a Hankel matrix, specifically $\mathcal{H}\boldsymbol{x} = \boldsymbol{X}^e$. Furthermore, given that $\boldsymbol{x}$ is a time series obeying an LRF defined above, $\mathcal{H}\boldsymbol{x}$ is a matrix with a rank of at most r. Time series that satisfy a LRF have broad applications, including economic forecasting, epidemiology, and control systems, as noted in [14, 87]. Unlike the ANM approach, the EMaC methodology is not limited to the SSS model but extends to other structured matrix completions, such as Circulant, Toeplitz, and Hankel matrices.

2.4 Spectral Compressed Sensing Optimization Formulation

Drawing from the preceding analysis, we can formulate the SCS problem as the following nonconvex optimization problem.

2.4.1 Denoising in Unweighted Array Norm

Mirroring the low-rank Hankel matrix approximation methodologies [88, 89], the Noisy-EMaC problem is formulated as:

$$\text{minimize} \quad \|\mathcal{H}\boldsymbol{x}\|_* \tag{2.31}$$

$$\text{subject to} \quad \|\mathcal{H}P_\Omega(\boldsymbol{x}-\boldsymbol{s})\|_F \le \epsilon, \tag{2.32}$$

where $\boldsymbol{s}, \boldsymbol{x} \in \mathbb{C}^n$. This problem can be addressed by SVT, minimizing $\|\mathcal{H}\boldsymbol{x}\|_* + \lambda \|\mathcal{H}P_\Omega(\boldsymbol{x}-\boldsymbol{s})\|_F$ [2, 84, 88]. It bears resemblance to the Cadzow algorithms [54], which use alternating projection between the nearest low-rank matrix and the nearest Hankel matrix. However, the Hankel operator induces varying weights across the vector $P_\Omega(\boldsymbol{x}-\boldsymbol{s})$, leading to a weighted least-squares problem. To circumvent this, an unweighted Cadzow method has been proposed [64], enhancing recovery accuracy through additional projections.

As an alternative, we present the Noisy-EMaC in an array form:

$$\text{minimize} \quad \|\mathcal{H}\boldsymbol{x}\|_* \tag{2.33}$$

$$\text{subject to} \quad \|P_\Omega(\boldsymbol{x}-\boldsymbol{s})\|_F \le \epsilon. \tag{2.34}$$

Though this approach improves reconstruction accuracy, it comes at the cost of increased computational effort and potentially slower convergence [64, 86].

This SCS formulation, particularly the Noisy-EMaC approach, provides a robust framework for

addressing denoising in SCS, especially in cases involving unweighted array norms.

2.4.2 Nonconvex Approaches

While enhanced Hankel matrix approaches, such as the SVT, benefit from the Hankel structure for acceleration, they still require a full SVD in each iteration [2, 90]. This significant drawback limits their scalability as the problem size increases. Originating from their dependency on matrix factorizations, these methods are typically efficient only for medium-sized problems [91]. To augment efficiency, several studies have proposed non-convex approaches utilizing truncated SVD [56–58, 61, 86], where the modeling order is assumed to be known in advance. Recent research indicates that nonconvex formulations can offer advantages in specific signal processing contexts, providing solutions with superior performance [45, 92], and leading to computationally manageable problems [93]. In these scenarios, the concerns regarding spurious local minima are either mitigated or non-existent. Consequently, with suitable initializations, local optimization methods often converge to the global minima [94].

Therefore, rather than the nuclear norm minimization of $\boldsymbol{X}^e$, the rank constriant is introduced[3], denoted as $\delta(\text{rank}(\boldsymbol{X}^e) \leq r)$, where $\delta(\cdot)$ is the indicator function for rank-constrained matrices, valued at zero if true and infinity otherwise. Moreover, the $\boldsymbol{X}^e = \mathcal{H}\boldsymbol{x}$ here is introduced due to the lifting technique, which refers to the process of transforming a problem into a higher-dimensional space, often with the aim of making the problem more tractable or solvable with the available algorithms [95]. For example, the matrix pencil approach [31] has been utilized to tackle the nonlinearity and nonconvexity in line spectral estimation.

Conclusively, the rank-constrained nonconvex problem is formulated as:

$$\min_{\boldsymbol{X}^e, \boldsymbol{x}} \ \delta(\text{rank}(\boldsymbol{X}^e) \leq r) + \frac{1}{2}\|\boldsymbol{s} - \mathcal{P}_\Omega \boldsymbol{x}\|_2^2 \quad \text{s.t. } \boldsymbol{X}^e = \mathcal{H}\boldsymbol{x}, \tag{2.35}$$

This reformulation towards nonconvex methods paves the way for more efficient solutions in

[3]The model order can be either estimated in advance or approximated through rank-increasing heuristics [56, 57].

large-scale SCS problems, leveraging the rank minimization to mitigate computational challenges.

2.4.3 Proximal Gradient Descent

Given the focus on solving the nonsmooth objective function delineated in (2.35), it is pertinent to explore recent advancements in numerical optimization, particularly for structured nonsmooth optimization problems. Traditional methodologies like Newton-type methods are not directly suitable for these scenarios. Consequently, there has been a resurgence in the development and application of splitting algorithms [6, 96, 97]. These first-order algorithms are adept at minimizing nonsmooth cost functions while maintaining minimal memory requirements, making them ideal for large-scale problems. Their primary limitation, however, lies in their relatively slow convergence rate.

In response to this limitation, significant research efforts have been dedicated to optimizing and accelerating these algorithms. Notably, the PG algorithm, also known as forward-backward splitting [98] or iterative shrinkage-thresholding algorithm (ISTA) [99], and the ADMM [79] are among the most prominent splitting algorithms. The PG algorithm's earliest known acceleration can be traced back to [100], leading to what is now called the fast proximal gradient algorithm or fast iterative shrinkage-thresholding algorithm (FISTA) [99].

More recent innovations in accelerating the PG algorithm include the variable metric forward-backward algorithm [101–104] and the integration of quasi-Newton methods [105–108]. These latter methods effectively utilize Hessian information to enhance the optimization process.

In summary, the development of these advanced algorithms reflects a significant step forward in efficiently addressing large-scale structured nonsmooth optimization problems, particularly in the context of SCS.

2.5 Challenges and Motivations

In light of the preceding discussions, the challenges and motivations guiding this thesis can be summarized as follows:

1. **Incorporating Damping Factors:** *super-resolution* SCS algorithms that leverage multi-block Hankel matrices need to address the damping factor present in signal models, which is a critical component in various applications, such as MRI.

2. **Addressing Bias from Hankel Structure:** It is noteworthy that the enhanced matrix approaches tend to consider elements repetitively due to the Hankel operator. This inherent bias should be effectively addressed, especially in noisy scenarios, to ensure accurate signal recovery.

3. **Handling Large-Scale Data:** Efficient algorithms, such as SVT method, are necessary to manage large-scale data sets commonly encountered in real-world applications. Such algorithms must be capable of processing high volumes of data without significant computational bottlenecks.

4. **Overcoming Slow Convergence:** First-order methods often face challenges related to slow convergence speeds, particularly in demanding situations characterized by low sampling rates and substantial noise. Developing strategies to expedite convergence while maintaining accuracy is crucial.

These challenges form the basis of our research focus, driving the development of improved algorithms and techniques in the field of *super-resolution* SCS.

2.6 Methodologies

In this section, we briefly outline the *super-resolution* SCS techniques for exponential signals as follows:

- N-D exponential signal
  - Undamped complex sinusoid recovery
    - Approximately on the grid
      - Compressed sensing [21]
    - gridless
      - Total variation minimization [4]
      - Atomic norm minimization [5, 9, 17]
      - Enhanced matrix completion [1, 27-32,35]
  - Damped complex sinusoid recovery
    - Enhanced matrix completion [1, 27-32,35]

(2.36)

The thesis title reflects the main motivations and advantages from an algorithmic perspective, focusing on three key aspects: **multi-block Hankel**, **unweighted**, and **fast proximal gradient methods**. Below, we list and elaborate on these features to address the challenges identified earlier:

1. **Multi-block Hankel Matrix Approach:** This approach is chosen due to its storage and computational efficiency, exploiting the low rank and Hankel structure. It offers superior resolution and generalizes well to handle damped signals. Variations of nonconvex modes are also considered. The computational package for the multi-block Hankel matrix is introduced in Section 3.

2. **Unweighted Array Norm:** The unweighted array norm replaces the traditional enhanced Hankel matrix completion for improved reconstruction. The slow convergence due to size-dependent Lipschitz constants is mitigated through a modified proximal gradient

strategy.

3. **Proximal Gradient Method Variations:** To accelerate convergence, several variations of proximal gradient methods are introduced. These include:

   (a) Low-rank projection to further decrease the objective function.

   (b) Hankel space projection for larger step sizes, inversely proportional to the sampling rate.

   (c) Incorporation of Hessian information to address the large conditional number challenge.

This methodology section thus sets the foundation for addressing the SCS challenges with efficient, advanced algorithmic strategies.

Part II

Methodologies

Chapter 3

Fast Computation Tools for Multi-block Hankel Matrices

Building upon the discussion in Chapter 2, where we highlighted the efficiency of the enhanced multiblock Hankel matrix approach in SCS, this chapter introduces a Julia package designed to enhance these computations. The package comprises several key tools, each catering to different aspects of Hankel matrix manipulation:

1. **Hankel Mapping:** This function maps an array to a Hankel matrix structure, defined by the array $\boldsymbol{s}$ and pencil parameter index $\boldsymbol{p}$:

   ```
   H = Hankel(s,p)
   ```

   The enhanced matrix is efficiently stored as an array, avoiding explicit construction of the Hankel matrix for computational and storage efficiency. Scalar multiplication, conjugation, addition, subtraction, and transpose operations are implemented in dense form, requiring only $\mathcal{O}(n)$ flops. The full Hankel matrix is not generated unless explicitly invoked:

   ```
   fullHankel(H)
   ```

2. **Hankel Matrix-Vector Multiplication:** Efficiently computes the product of a $p \times q$ $(p + q = n + 1)$ Hankel matrix and a vector using FFT, requiring $\mathcal{O}(n \log n)$ flops [51].

```
H*x
```

3. **Truncated Singular Value Decomposition (TSVD) of Hankel Matrices:** This tool efficiently computes the r-truncated SVD of a Hankel matrix, utilizing the FFT and implicitly restarted Lanczos bidiagonalization (irlb) method [5], with $\mathcal{O}(rn \log n)$ flops. It offers several advantages over other methods:

   (a) Enhanced efficiency tailored for TSVD.

   (b) Initialization for FFT planned once to minimize repetitive calculations.

   (c) Inclusion of modified SVD functions widely used in CS.

```
# Standard r-largest truncated SVD
U1, S1, V1 = irlb(H,r,tol,iter,ncv)
# Linear combination of the Hankel and \\
# a low-rank matrix X=U1*S1*transpose(V1)
U2, S2, V2 = irlblr(H,r,tol,iter,ncv,U1*S1,V1)
# Iterative TSVD based on Riemann manifold U1 and V1
U2, S2, V2 = riemannsvd(U1,V1,H)
```

4. **Penrose Pseudoinverse for Hankel Operator $\mathcal{H}^\dagger$:** Implements the left inverse operator of $\mathcal{H}$, transforming $p \times q$ matrices into vectors of length n. Efficiently executed using FFT for matrices where $r \ll n$, with $\mathcal{O}(r^2 n + rn \log n)$ flops [51]. For a matrix $\boldsymbol{X} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^*$, the operation $\mathcal{H}^\dagger \boldsymbol{X}$ is performed as:

```
ph = projectlowHankel(U,S,V,n,p)
```

These tools are integral to the implementation of the algorithms proposed in subsequent chapters. Detailed descriptions of these tools are provided in the following sections. The package is available at `https://github.com/xiyao65/multiblockHankelMatrices.jl`.

3.1 Matrix-vector Multiplication via FFT

To effectively leverage the FFT [51], we exploit the structure of the Hankel matrix by transforming it into a circulant matrix $\boldsymbol{C}$. A circulant matrix is a square matrix where each row is a cyclic shift to the right of the previous row.

Given a Hankel matrix $\boldsymbol{H} \in \mathbb{C}^{p\times q}$ with its first column $\boldsymbol{c} = [c_1, \ldots, c_p]^T$ and last row $\boldsymbol{r} = [r_1, r_2, \ldots, r_q]$ (where $p \geq q$, $r_1 = c_p$ and $p + q = n + 1$), the first column of the corresponding circulant matrix $\boldsymbol{C} \in \mathbb{C}^{n\times n}$ is defined as:

$$\hat{\boldsymbol{c}} = [c_1, \ldots, c_p, r_2, \ldots\ r_q]^T.$$

For a vector $\boldsymbol{x}$ with q elements, the matrix-vector product $\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x}$ is equal to the last p elements of $\hat{\boldsymbol{y}}$, where

$$\hat{\boldsymbol{y}} = \boldsymbol{C}\hat{\boldsymbol{x}}, \quad \text{with} \quad \hat{\boldsymbol{x}} = [x_q, x_{q-1}, \ldots, x_1, \underbrace{0, \ldots, 0}_{p-1}]^T.$$

The circulant matrix $\boldsymbol{C}$ can be spectrally factorized using the Fourier matrix $\boldsymbol{F}$, leading to the following relation:

$$\boldsymbol{C}\hat{\boldsymbol{x}} = \text{inv}(\boldsymbol{F})\text{diag}(\boldsymbol{F}\hat{\boldsymbol{c}})\boldsymbol{F}\hat{\boldsymbol{x}},$$

which implies

$$\hat{\boldsymbol{y}} = \text{ifft}(\text{fft}(\hat{\boldsymbol{c}}) * \text{fft}(\text{rev}(\hat{\boldsymbol{x}}))).$$

This approach to matrix-vector multiplication via FFT significantly enhances computational efficiency, particularly for large-scale data handling in SCS.

The application of FFT for circulant matrices is well-established and extends efficiently to multi-level structures [81, 109]. In the context of this thesis, a 1-D FFT is specifically employed for multi-block Hankel (MBH) matrix-vector multiplication. This approach is advantageous for

its minimal memory requirements and enhanced computational efficiency [3].

The implementation of MBH matrix-vector multiplication is streamlined in Julia, utilizing the native array manipulation functionalities. The operation of vectorization, which reshapes an array into a one-dimensional column vector, is a built-in feature in the Julia.Base package. To circumvent the computational overhead associated with recursive operations as described in [3], we adopt pad and extract operations in array form. The following algorithm outlines this efficient process:

Algorithm 2 MBH Matrix-Vector Multiplication [3]

Input: The multi-dimensional array: $\boldsymbol{s}$.

$\hat{\boldsymbol{y}} \leftarrow \text{ifft}(\text{fft}(\text{vec}(\boldsymbol{s})) * \text{fft}(\text{rev}(\text{pad}(\hat{\boldsymbol{x}}))))$;

$\boldsymbol{y} \leftarrow \text{extract}(\hat{\boldsymbol{y}})$;

Return $\boldsymbol{y}$;

This algorithm effectively leverages FFT for MBH matrix-vector multiplication, streamlining the computation and ensuring high performance, especially in large-scale SCS scenarios.

pad($\boldsymbol{x}$) To accommodate the transformation of a circulant matrix with size $(n_1 \dots n_k)$-by-$(n_1 \dots n_k)$, the vector $\boldsymbol{x}$ of dimensions $(q_1 \dots q_k)$ needs to be extended into an array of size $(n_1 \times \cdots \times n_k)$. This extension involves padding with zeros and is executed in three steps:

1. Reverse the vector $\tilde{\boldsymbol{x}}$.
2. Reshape $\tilde{\boldsymbol{x}}$ into $\bar{\boldsymbol{X}} \in \mathbb{C}^{q_1 \times q_2 \cdots \times q_k}$.
3. Expand $\bar{\boldsymbol{X}}$ into $\hat{\boldsymbol{X}} \in \mathbb{C}^{n_1 \times n_2 \cdots \times n_k}$ by padding zeros at the end of each dimension.

extract($\hat{\boldsymbol{y}}$) While leveraging 1-D FFT for computational efficiency, it remains essential to extract the vector $\boldsymbol{y}$ of dimensions $(p_1 \dots p_k)$ from the larger vector $\hat{\boldsymbol{y}}$ of size $(n_1 \dots n_k)$. This extraction process is conducted as follows:

1. Reshape the reversed vector of $\hat{\boldsymbol{y}}$ into $\bar{\boldsymbol{Y}} \in \mathbb{C}^{n_1 \times n_2 \cdots \times n_k}$.

2. From $\bar{\boldsymbol{Y}}$, extract $\dot{\boldsymbol{Y}} \in \mathbb{C}^{p_1 \times p_2 \cdots \times p_k}$ starting from the beginning of each dimension.

3. Reverse the vectorization of $\dot{\boldsymbol{Y}}$ to obtain $\boldsymbol{y}$.

Consequently, the MBH matrix-vector multiplication method achieves a computational complexity of $\mathcal{O}(n_1 n_2 \dots n_k \log(n_1 n_2 \dots n_k))$. This represents an improvement in efficiency compared to the complexity of $\mathcal{O}(n_1 n_2 \dots n_k \log(n_1) \log(n_2) \cdots \log(n_k))$, which is more typical of multi-level FFT approaches. By consolidating the logarithmic terms, the 1-D MBH approach streamlines the process, ensuring faster and more efficient computation suitable for high-dimensional SCS tasks.

The efficiency and adaptability of the fast MBH matrix-vector multiplication method are also influenced by the choice of the pencil parameter $\boldsymbol{p}$, which is used to choose makes the enhanced Hankel matrix close to square. Therefore, this parameter plays a crucial role in determining the structure and efficiency of the Hankel matrix computations. Notably, the principles underpinning this methodology are not limited to Hankel matrices alone. They can be similarly applied to other structured matrices, enhancing the computation in various contexts. For instance, this approach can be effectively adapted for Toeplitz matrices, which are prevalent in signal processing and control theory. Additionally, the methodology is highly relevant to the multiple measurement vector (MMV) model, which is central to many applications in array signal processing, medical imaging, and geosciences. The ability to apply this fast MBH matrix-vector multiplication technique across different structured matrices greatly enhances its utility, making it a versatile tool in computational mathematics and engineering.

Toeplitz Matrix The development of fast matrix-vector multiplication for structured matrices such as Toeplitz matrices can be understood by examining their relationship with circulant matrices. This approach is similar to that used for Hankel matrices, with subtle differences in the padding and extraction processes.

Consider a simple example with a vector $\boldsymbol{s} = [s_1, s_2, s_3]$. The corresponding circulant matrix $\boldsymbol{C}_s$, Hankel matrix $\boldsymbol{H}_s$, and Toeplitz matrix $\boldsymbol{T}_s$ can be represented as follows:

$$\boldsymbol{C}_s = \left[\begin{array}{cc|c} s_1 & s_3 & s_2 \\ \hline s_2 & s_1 & s_3 \\ s_3 & s_2 & s_1 \end{array}\right], \quad \boldsymbol{H}_s = \begin{bmatrix} s_1 & s_2 \\ s_2 & s_3 \end{bmatrix}, \quad \boldsymbol{T}_s = \begin{bmatrix} s_2 & s_1 \\ s_3 & s_2 \end{bmatrix}.$$

It can be observed that $\boldsymbol{H}_s$ is essentially a reversed row embedded in the lower left side of $\boldsymbol{C}_s$, which aligns with the structure of $\boldsymbol{T}_s$. This observation allows for the adaptation of a fast matrix-vector multiplication approach for $\boldsymbol{T}_s$. Unlike the reversal required before padding in Hankel matrices, padding can be applied directly to Toeplitz matrices. Moreover, our method, which relies on padding and extracting operations on the vector, provides an intuitive and flexible solution for handling arbitrary large-dimensional structured matrices and their variants. This approach, though inspired by the circulant matrix framework as detailed in [81], offers a more straightforward implementation for Toeplitz matrices.

MMV Model Matrix-vector multiplication within the MMV model can also be formulated as a composition of multiple Hankel matrix-vector multiplications. This is exemplified as follows:

$$[\boldsymbol{H}_1|\boldsymbol{H}_2|\boldsymbol{H}_3] \begin{bmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \\ \boldsymbol{x}_3 \end{bmatrix} = [\boldsymbol{H}_1 * \boldsymbol{x}_1|\boldsymbol{H}_2 * \boldsymbol{x}_2|\boldsymbol{H}_3 * \boldsymbol{x}_3],$$

where $\boldsymbol{H}_i$ $(i = 1, 2, 3)$ represent individual Hankel matrices. In this model, the overall matrix-vector multiplication result is essentially a composition of the products of these Hankel matrices with their corresponding vectors. This approach effectively treats the MMV model as a reduced form of a level-2 Hankel matrix, allowing each multiplication to be handled separately and efficiently. The decomposition of the overall multiplication into separate Hankel matrix-vector operations simplifies the computation, making the MMV model a powerful tool for processing multiple measurement vectors in a structured and computationally efficient manner.

3.2 Truncated Singular Value Decomposition of Hankel Matrices

The Lanczos process computes the partial tridiagonalization of a symmetric matrix based on matrix-vector multiplication, where the largest r eigenvalues and eigenvectors emerge relatively early in the process, typically within m steps [51]. Considering this efficient eigenvalue decomposition of symmetric matrices, a symmetric intermediate for SVD can be constructed as follows:

1. $$\boldsymbol{Z} \coloneqq \begin{bmatrix} 0 & \boldsymbol{H} \\ \boldsymbol{H}^{\mathsf{H}} & 0 \end{bmatrix}$$

   This associated symmetric matrix is utilized in ARPACK [110] with implicitly restarted Arnoldi methods [111, 112], which involve a larger size of Lanczos vectors.

2. The matrix $\boldsymbol{H}^{\mathsf{H}}\boldsymbol{H}$ disrupts the Hankel structure and increases error due to the squaring of the condition number. Another drawback is that if the special structure is to be preserved, two separate matrix-vector multiplications need to be conducted.

Rather than using the tridiagonalization Lanczos process, Golub-Kahan bidiagonalization [4] demonstrates an efficient method for computing the singular value decomposition (SVD). For large matrices $\boldsymbol{H} \in \mathbb{C}^{p\times q}$, unless the partial bidiagonalization step is kept small, the storage requirements and computational costs become substantial, especially due to the reorthogonalization process necessitated by round-off errors. However, with a small value of m, the Ritz approximations may not be accurate. To address this, various restarted approaches have been proposed.

One such approach involves a sequence of Lanczos partial bidiagonalizations with fixed, relatively small m, but with improved initial vectors. Sorensen [112] developed implicit Arnoldi methods, akin to QR algorithms with shifts. While this shift-and-invert approach is effective for small-sized

problems, it can become untenable for larger problems due to excessive memory and execution time requirements. Additionally, numerical instability can arise from propagated round-off errors.

To enhance robustness, Krylov subspaces are augmented with certain Ritz vectors in thick-restarted tridiagonalization methods [113]. Stewart [114] proposed an alternative restart procedure based on Schur decomposition within Krylov subspaces. The thick-restarted bidiagonalization method [5] is an equivalent but simpler implementation of the Krylov-Schur approach [115]. Moreover, numerical simulations indicate that the implicitly restarted Lanczos bidiagonalization (irlb) method converges faster than the existing KrylovKit.jl package, which adopts the Krylov-Schur methods.

As previously mentioned, the Golub-Kahan Bidiagonalization (GKB) process is utilized to extract information about desired singular vectors/values efficiently. The GKB process is detailed in Algorithm 3.

Algorithm 3 Golub-Kahan Bidiagonalization (GKB) [4]

Input: A matrix $\boldsymbol{H} \in \mathbb{C}^{p\times q}$, and the size of Krylov space: m.

Set $\boldsymbol{v}_0 \in \mathbb{C}^q$ as the initial vector, $k \leftarrow 0$, $\beta_0 \leftarrow 1$, $\boldsymbol{u}_0 \leftarrow \boldsymbol{0}$.
while $\beta_k \neq 0$ & $k \leq m$ **do**
  $\boldsymbol{v}_{k+1} \leftarrow \boldsymbol{p}_k/\beta_k$.
  $k \leftarrow k+1$.
  $\boldsymbol{r}_k \leftarrow \boldsymbol{H}\boldsymbol{v}_k - \beta_{k-1}\boldsymbol{u}_{k-1}$.
  $\alpha_k \leftarrow \|\boldsymbol{r}_k\|_2$.
  $\boldsymbol{u}_k \leftarrow \boldsymbol{r}_k/\alpha_k$.
  $\boldsymbol{p}_k \leftarrow \boldsymbol{H}^{\mathsf{H}}\boldsymbol{u}_k - \alpha_k\boldsymbol{v}_k$.
  $\beta_k \leftarrow \|\boldsymbol{p}_k\|_2$.
end while

Output: Matrices $\boldsymbol{H}\boldsymbol{V}_m = \boldsymbol{U}_m\boldsymbol{B}_m$ and $\boldsymbol{H}^{\mathsf{H}}\boldsymbol{U}_m = \boldsymbol{V}_m\boldsymbol{B}_m^T + \boldsymbol{p}_m\boldsymbol{e}_m^T$, with $\boldsymbol{V}_m = [\boldsymbol{v}_1|\cdots|\boldsymbol{v}_m]$, $\boldsymbol{U}_m = [\boldsymbol{u}_1|\cdots|\boldsymbol{u}_m]$ and $\boldsymbol{B}_m$ as defined in (3.1).

$$\boldsymbol{B}_m = \begin{bmatrix} \alpha_1 & \beta_1 & \dots & \dots & 0 \\ 0 & \alpha_2 & \beta_2 & \dots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ \vdots & & 0 & \alpha_{m-1} & \beta_{m-1} \\ 0 & \dots & 0 & 0 & \alpha_m \end{bmatrix} \tag{3.1}$$

After executing the Golub-Kahan Bidiagonalization (GKB) process with m steps, typically set as $m = 2r$ [5], the accuracy of the approximated r-triplets of $\boldsymbol{H}$ is checked against the criterion $\beta_m \|\boldsymbol{e}_m^T \boldsymbol{u}_j^{\boldsymbol{B}_m}\| \leq \delta \|\boldsymbol{H}\|$, with δ as the tolerance. If convergence is not achieved, iterations need to be implicitly restarted to enhance accuracy. The dilemma lies in maintaining relevant spectral information while continuing the GKB process.

Similar to restarted Arnoldi methods, the GKB output is reduced to a smaller factorization:

$$\boldsymbol{H}\tilde{\boldsymbol{V}}_{r+1} = \tilde{\boldsymbol{U}}_{r+1}\tilde{\boldsymbol{B}}_{r+1}, \quad \boldsymbol{H}^{\mathsf{H}}\tilde{\boldsymbol{U}}_{r+1} = \tilde{\boldsymbol{V}}_{r+1}\tilde{\boldsymbol{B}}_{r+1} + \tilde{\beta}_{r+1}\tilde{\boldsymbol{v}}_{r+2}\boldsymbol{e}_r^T, \tag{3.2}$$

where $\tilde{\boldsymbol{U}}_r, \tilde{\boldsymbol{v}}_r$, and $\tilde{\boldsymbol{B}}_r$ collect the desired singular information. To continue the GKB process, we determine $\tilde{\boldsymbol{V}}_{r+1}, \tilde{\boldsymbol{U}}_{r+1}, \tilde{\boldsymbol{B}}_{r+1}$, and $\tilde{\boldsymbol{v}}_{r+2}$ as in (3.3):

$$\tilde{\boldsymbol{V}}_{r+1} = [\boldsymbol{q}_1, \boldsymbol{q}_2, \ldots, \boldsymbol{q}_r, \boldsymbol{v}_{m+1}], \quad \tilde{\boldsymbol{U}}_{r+1} = [\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_r, \tilde{\boldsymbol{u}}_{r+1}], \tag{3.3}$$

where $\boldsymbol{q}_i = \boldsymbol{V}_m \boldsymbol{y}_i$, $\boldsymbol{p}_i = \boldsymbol{U}_m \boldsymbol{x}_i$, and $\boldsymbol{x}_i, \boldsymbol{y}_i$ are the left and right singular vectors of $\boldsymbol{B}_m$. The orthogonality of $\tilde{\boldsymbol{U}}_{r+1}$ is maintained by setting $\tilde{\boldsymbol{u}}_{r+1} = \text{normalize}\,(\boldsymbol{H}\boldsymbol{v}_{m+1} - \sum_{i=1}^{r} \rho_i \boldsymbol{p}_i)$.

The final factorization obtained is:

$$\boldsymbol{H}\tilde{\boldsymbol{V}}_m = \tilde{\boldsymbol{U}}_m\tilde{\boldsymbol{B}}_m, \quad \boldsymbol{H}^{\mathsf{H}}\tilde{\boldsymbol{U}}_m = \tilde{\boldsymbol{V}}_m\tilde{\boldsymbol{B}}_m + \tilde{\beta}_m\tilde{\boldsymbol{v}}_{m+1}\boldsymbol{e}_m^T, \tag{3.4}$$

with $\tilde{\boldsymbol{B}}_m$ defined as:

$$\tilde{\boldsymbol{B}}_m = \begin{bmatrix} \sigma_1 & 0 & \ldots & 0 & \rho_1 & \ldots \\ \vdots & \ddots & \ddots & \ldots & \vdots & \\ 0 & \ldots & \sigma_r & \rho_r & 0 & \ldots \\ & & & \alpha_{r+1} & \beta_{r+1} & \ldots \\ & & & & \ddots & \ddots \\ 0 & & & & & \alpha_m \end{bmatrix}.$$

This procedure is repeated until convergence or the outset of another iteration.

Although $\tilde{\boldsymbol{B}}_m$ is no longer a bidiagonal matrix, the SVD is just a slight disadvantage due to the

small size of m. Finally, the whole algorithm of irbl is given below in Algorithm 4:

Algorithm 4 Truncated SVD by Implicitly Restarted Lanczos Bidiagonalization [5]

Input: a matrix $\boldsymbol{H} \in \mathbb{C}^{p\times q}$, the size of Krylov space m, number of desired singular triplets r, tolerance δ, machine epsilon ϵ.

Set $\boldsymbol{v}_0 \in \mathbb{C}^q$ as the initial vector, $k \leftarrow 0$, $\beta_0 \leftarrow 1$, $\boldsymbol{u}_0 \leftarrow \boldsymbol{0}$.
Compute the partial Lanczos bidiagonalization by Algorithm 3.
Compute the singular value decomposition of $\boldsymbol{B}_m$.
Check convergence: if all r desired singular triplets satisfy $\beta_m \|\boldsymbol{e}_m^T \boldsymbol{u}_j^{\boldsymbol{B}_m}\| \leq \delta \|\boldsymbol{H}\|$, then exit.
Compute the augmenting vectors:
while not converged **do**
Determine the matrices $\boldsymbol{V} \coloneqq \tilde{\boldsymbol{V}}_{r+1}, \boldsymbol{U} \coloneqq \tilde{\boldsymbol{U}}_{r+1}$ by (3.3).
Determine $\boldsymbol{B} \coloneqq \tilde{\boldsymbol{B}}_{r+1} \coloneqq \begin{bmatrix} \sigma_1 & & \boldsymbol{0} & \rho_1 \\ & \ddots & & \vdots \\ & & \sigma_r & \rho_r \\ \boldsymbol{0} & & \dots & \alpha_{r+1} \end{bmatrix}$.
Append $\boldsymbol{U}, \boldsymbol{V}, \boldsymbol{B}$ to $\boldsymbol{U}_m, \boldsymbol{V}_m, \boldsymbol{B}_m$.
Calculate the SVD of $\boldsymbol{B}_m$.
end while

Output: Computed set of approximate singular triplets $\{\sigma_j, \boldsymbol{u}_j, \boldsymbol{v}_j\}_{j=1}^r$ of $\boldsymbol{H}$.

Remark 5. *In practical implementation of Algorithm 4, orthogonalization is required to compensate for round-off errors. The computational complexity is observed to be $\mathcal{O}(rM)$ with M being the flops of matrix-vector multiplication. Combined with fast matrix-vector multiplication of the MBH matrix, the TSVD is achieved with $\mathcal{O}(rn\log(n))$ flops. The size of Krylov subspaces is set to $2r$ by default. An early stopping mechanism is added to monitor the norm of the residual vector for efficiency.*

Numerical Results of TSVD via irlb with FFT

Numerical simulations demonstrate the advantages of the truncated singular value decomposition (TSVD). For randomly generated Hankel matrices, TSVD is computed for the largest 10 singular triplets with a specified level of precision. The primary criterion for comparison is execution time. According to [56], we set $\boldsymbol{p} = \lfloor \frac{\boldsymbol{n}}{2} \rfloor + \boldsymbol{1}$ to make the Hankel matrix approximately square.

Benefits of TSVD First, we illustrate the efficiency of TSVD using the restarted approach, compared to the full SVD without FFT, as shown in Table 3.1. The tolerance is set to 1×10^{-10},

and the deviation from the true singular values computed by SVD is at most 1×10^{-13} in the simulations. The size of the Krylov subspace is 25. The ratio of svd/irlb is also presented.

Table 3.1: Running time comparison of irlb and full svd.

Dimension of Data	**irlb (s)**	**svd (s)**	**svd/irlb Ratio**
(5000)	$5.01e-1$	9.8	19.42
$(128, 128)$	1.96	43.89	22.39
$(30, 30, 30)$	2.47	29.19	11.79

Benefits of irlb The numerical experiments in this part briefly compare the proposed package tool against existing competitive ones (two packages in the Julia community, KrylovKit and Arpack). First, we construct an abstract MBH matrix with fast matrix-vector multiplication. There is no on-the-shelf package available for MBH. Our irlb package is tailored for MBH matrices, meaning we can have a multidimensional array as input to utilize the FFT. The number of iterations (NI) and matrix-vector multiplication (NMV) are also considered vital criteria for comparison. Additionally, Arpack.svds applies the routine to the matrix $H^{\mathsf{H}}H$, which is efficient for small dimensions but suffers when the matrix is relatively large and breaks the Hankel structure. Therefore, we will only compare with the KrylovKit package with similar parameter settings. The results presented in Table 3.2 for different sizes of the MBH matrix display the benefit of irlb in various examples. It is observed that the irlb method has fewer iterations than the Krylov-Schur methods. Although we have not optimized the orthogonalization process as has been done in KrylovKit, the benefits are still obvious, especially for large-scale data.

Table 3.2: Comparison of irlb with KrylovKit.

Method	**Dimension**	**NI**	**NMV**	**Time (s)**
irlb	(5000)	7	184	1.72e-1
KrylovKit	(5000)	10	204	1.66e-1
irlb	(256, 256)	6	182	1.82
KrylovKit	(256, 256)	10	210	1.51
irlb	(50, 50, 50)	9	232	3.51
KrylovKit	(50, 50, 50)	14	274	3.62
irlb	(30, 30, 30, 30)	11	294	33.21
KrylovKit	(30, 30, 30, 30)	19	376	49.35

3.3 Fast Penrose Pseudoinverse of Multi-block Hankelization

In this section, we introduce an efficient implementation for the Penrose pseudoinverse of Hankelization on a low-rank matrix. This mapping from a low-rank matrix to a MBH matrix is crucial in SSS recovery and other low-rank MBH Hankel completion problems. Our implementations, which apply one-dimensional fast Fourier transform (1D FFT) directly, enjoy reduced execution time, filling a gap in existing computational packages for large-scale and multiblock Hankelization.

For a given matrix $\boldsymbol{H} \in \mathbb{C}^{p_1 \cdots p_k \times q_1 \cdots q_k}$, the array $\mathcal{H}^\dagger \boldsymbol{H} \in \mathbb{C}^{n_1 \times n_2 \cdots \times n_k}$ is obtained by averaging each skew-diagonal of the submatrix. This is akin to the conjugate operator of Hankelization, $\mathcal{H}^*$, which sums the elements according to the MBH structure. The general computational complexity for this operation is $\mathcal{O}(n_1^2 \cdots n_k^2)$. In our applications, such as Cadzow filtering, this operator acts on a low-rank matrix with Singular Value Decomposition (SVD) represented as $\boldsymbol{H} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^*$, leading to the following formulation:

$$\mathcal{H}^\dagger(\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^*) = \sum_{l=1}^{r} \boldsymbol{\Sigma}(l,l) \left(\mathcal{H}^\dagger(\boldsymbol{U}(:,l))(\boldsymbol{V}(:,l))^*\right). \tag{3.5}$$

Applying fast convolution in equation (3.5), each term $\mathcal{H}^\dagger(\boldsymbol{U}(:,k))(\boldsymbol{V}(:,k))^*$ is computed with $\mathcal{O}(n\log(n))$ flops. Therefore, the total computation for $\mathcal{H}^\dagger(\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^*)$ requires $\mathcal{O}(rn\log(n))$ flops. As the enhanced MBH Hankel matrix is close to a square matrix, the FFT approach for convolution is preferred over the overlap-save method. Inspired by the 1D FFT for matrix-vector multiplication, the 1D convolution for the Penrose pseudoinverse of multiblock Hankelization achieves a computational complexity of $\mathcal{O}(n_1 \cdots n_k(\log(n_1) + \cdots + \log(n_k))$ flops, rather than $\mathcal{O}(n_1 \cdots n_k \log(n_1) \cdots \log(n_k))$ flops as required by multi-level convolutions.

Benefits of 1D FFT Numerical benefits of using 1D FFT, as opposed to the DSP.jl package, which adopted multidimensional FFT, for the Penrose pseudoinverse of rank-1 multi-block

Hankelization are demonstrated. The comparison is based on time consumption, with the ratio of time taken by 1D FFT to DSP.jl denoted as "1D/DSP". The results are presented in Table 3.3, showing the superior performance of 1D FFT in terms of computational efficiency.

Table 3.3: Running time (s) comparison of the 1D FFT with DSP.

Dimension of data	1D	DSP	DSP/1D
(5000)	1.02e-4	1.65e-4	1.62
(3, 7, 21, 7)	7.8e-5	1.91e-4	2.44
(7, 3, 5, 11, 3)	9.2e-5	1.78e-4	1.93

3.4 Conclusions

This chapter delves into advanced computational techniques optimized for MBH matrices, focusing on spectral compressive sensing and its applications.

1. **Fast Matrix-Vector Multiplication for MBH Matrices:** We introduce a 1D-FFT method for expediting matrix-vector multiplication in MBH matrices. This approach enhances flexibility and computational efficiency, especially in handling large-scale data.

2. **TSVD for MBH Matrices:** The TSVD, grounded in the Golub-Kahan bidiagonalization, is presented as a more computationally efficient alternative to the full SVD. It effectively reduces computational demands while maintaining critical spectral information.

3. **Fast Penrose Pseudoinverse for MBH Matrices:** We explore a technique to compute the Penrose pseudoinverse of MBH matrices using FFT. This method demonstrates superior computational speed, making it ideal for large-scale data processing and intricate signal recovery.

4. **Numerical Results and Demonstrated Efficiency:** Empirical evidence underlines the effectiveness of the proposed methods over conventional techniques. These empirical findings confirm the theoretical benefits and practical superiority of the methods, underscoring their relevance in real-world scenarios.

In conclusion, the chapter contributes significantly to signal processing and spectral compressive sensing by introducing efficient computational strategies for MBH matrices. These methods facilitate faster and more effective data processing in various applications, particularly where handling complex signal structures and large-scale data is paramount.

Chapter 4

Low-Rank Projected Proximal Gradient

This chapter introduces a PG-based approach for recovering spectrally sparse signals (SSSs) from partially observed, noisy data. The focus is on addressing challenges associated with moderate scale datasets and low sampling rates. To manage larger scale signals, we propose a nonconvex and nonsmooth objective function achieved through relaxing the low-rank constraint of the enhanced Hankel matrix. Traditional super-resolution methods for SSS recovery via enhanced Hankel matrices often encounter reconstruction inaccuracies, primarily due to unequally weighted norms and excessively lenient enforcement of the Hankel structure in noisy environments.

To tackle these issues, we have developed our optimization framework in vector form, incorporating an adjustable Hankel enforcement parameter. This framework is designed to resolve the weighted convergence problem observed in the Cadzow method. Specifically, we investigate the PG method with a monotone decreasing strategy to ensure convergence. A notable limitation of standard proximal PG methods in SCS is their relatively slow convergence speed, largely due to the application of unweighted norms in the optimization process. We address this limitation by introducing the low-rank projected proximal gradient (LPPG) method. This method is structured to converge efficiently to stationary points via a two-step iterative process. The initial step involves a modified PG approach that allows for a constant step size, independent of the signal size, significantly enhancing the gradient descent phase. The subsequent step involves a strategy of low-rank subspace projection, facilitating optimization within a fixed column and

row subspace of a low-rank matrix to further reduce the objective function.

Both stages of the LPPG method have been meticulously designed to leverage the inherent low-rank and Hankel structures of the problem, thereby boosting computational efficiency. Our numerical simulations demonstrate a marked improvement in both efficiency and recovery accuracy of the LPPG method compared to existing benchmark algorithms. This enhanced performance is especially evident in scenarios characterized by significant noise levels, underscoring the robustness of our method and its suitability for practical, larger scale SSS recovery tasks.

The remainder of this chapter is structured as follows: We commence with Section 4.1, providing an overview of related work in the field. Subsequently, in Section 4.2, we delve into our optimization formulation tailored for SCS, specifically focusing on one-dimensional frequency models. This section meticulously derives our objective function, employing a refined nonconvex relaxation approach to ensure precision. In Section 4.3, we introduce our proposed LPPG method. This method is innovatively designed to locate the critical points of our established formulation. We offer a detailed exposition of the LPPG algorithm, particularly emphasizing its two-step iterative process. Moreover, we discuss the convergence properties and the subgradient-based criteria for stopping the algorithm, providing insights into its theoretical underpinnings. Section 4.4 is dedicated to exploring the computational complexity and efficient implementation strategies for the LPPG method. Moving forward, Section 4.5 presents numerical validations of our method in various settings. Here, we compare the efficiency and accuracy of the LPPG method against existing benchmarks to demonstrate its effectiveness. For readers interested in the technical depth, proofs supporting our results are comprehensively detailed in Section 4.7. Lastly, the chapter concludes with Section 4.6, where we encapsulate our principal contributions and the implications of our findings.

4.1 Introduction

As we embark on discussing the methodology of SCS from this chapter, it is pertinent to briefly review the underlying motivations for the compactness, which will also be beneficial for

understanding Chapters 5 and 6.

4.1.1 Related Gridless Work

Convex Approach: As previously discussed in Chapter 2, the gridless methods, despite their effectiveness, encounter limitations when applied to larger scale CS problems. This is primarily due to the high computational complexity involved in solving the equivalent semi-definite programming (SDP). To address this challenge, Enhanced Matrix Completion (EMaC) [2] was proposed, which is an evolution of the traditional spectral estimation technique known as Matrix Enhancement and Matrix Pencil. EMaC introduces a low-rank Hankel matrix completion problem, leveraging the shift-invariance property and spectral sparsity, and is underpinned by the Vandermonde decomposition.

Subsequently, recovery approaches based on Hankel matrices gained popularity [45, 90], particularly for larger scale problems, owing to their computational efficiency. This efficiency stems from the fact that the entire matrix involved is Hankel structured, thereby facilitating efficient decomposition. An alternative strategy employed is the Burer-Monteiro heuristic [52, 53], where low-rank matrices are represented via a bilinear outer product. This approach circumvents the need for explicit matrix singular value decomposition (SVD).

However, it is important to note, as highlighted in the worst-case analysis by [45], that convex optimization approaches can encounter failure scenarios, even when only a single element is missing. This observation underscores the need for careful consideration in the application of these methods, especially in scenarios where data completeness cannot be guaranteed.

Non-Convex Approach: The use of nuclear norm-based methods in larger scale problems is often hindered by their requirement for full SVD of the enhanced matrix, which is computationally intensive. As a result, various non-convex methods [14, 54–58] have been developed to directly address low-rank Hankel matrix completion challenges. One such method is Cadzow's algorithm [14, 54], which utilizes alternating projections between sets of Hankel and low-rank matrices.

Though an additional Newton-type step was proposed in [59] to achieve quadratic convergence, leveraging common regularity intersection conditions [60, 61] of Cadzow's algorithm, its high computational demand often precludes practical applications. Another notable advancement is the incorporation of Riemannian optimization techniques [62], which involve adding a projection to the direct sum of column and row spaces [57,58]. An alternative strategy involves representing a low-rank matrix or tensor as an outer product of low-rank matrices, allowing for direct optimization on component matrices, thus bypassing the need for explicit matrix decomposition [55, 56].

Among these non-convex methods, Fast Iterative Hard Thresholding (FIHT) [57] and Projected Gradient Descent (PGD) [56] stand out due to their efficiency and convergence guarantees. These guarantees are based on the Hankel incoherence property [2], making these algorithms particularly effective in handling larger scale SCS problems.

In the realm of SSS recovery using enhanced low-rank matrix approximation, two primary challenges are encountered:

1) **Biased Weighted Norm:** The first challenge arises from the biased weighted norm, a consequence of repeated elements in the induced Hankel matrix. This bias can adversely affect the accuracy of the signal recovery process, skewing the results and leading to suboptimal reconstructions.

2) **Computational Complexity:** The second major challenge is the significant computational cost associated with solving structured low-rank approximation problems. Central to this issue is the condition number of the Hessian matrix associated with the quadratic terms in the objective function. This condition number is closely related to the Hankel enforcement parameter β and the size of the signal. A high condition number, especially with large values of β and signal size, leads to extremely slow convergence rates. This is particularly problematic when traditional methods, such as evaluating the rank function through SVD, are employed. The computational complexity of SVD, typically $\mathcal{O}(n^3)$, renders it impractical for larger scale problems, resulting in an unmanageably high number of iterations required for gradient-based methods, even in

scenarios with moderately sized problems.

Addressing these challenges is crucial to enhance the efficiency and accuracy of SSS recovery, especially in large-scale scenarios where computational resources and time are critical factors.

4.1.2 Contributions of LPPG

To overcome the aforementioned challenges, this study introduces a LPPG method, meticulously designed in two steps, to address a specially formulated objective function with equal weighted norms. This method showcases efficient convergence and enhanced performance in demanding situations, such as in the presence of heavy noise and at low sampling rates. The key contributions of this work are summarized as follows:

1. **An Accurate Formulation:** We have developed a formulation that aligns more closely with the original problem model in two crucial respects. First, unlike traditional Hankel-based algorithms, our approach represents spectrally sparse signals (SSS) in vector form rather than augmented matrix form, effectively reducing bias caused by differing weights in the matrix l_2 norm induced by the Hankel operator [14, 58]. This significantly enhances reconstruction accuracy in proportion to the signal size. Additionally, the introduction of a Hankel enforcement parameter in our unconstrained optimization framework allows for adjustments based on noise levels, thereby improving accuracy in highly noisy environments.

2. **Convergent Low-Rank Projected Proximal Gradient Algorithm:** The LPPG algorithm we propose efficiently converges to critical points, unrestricted by initialization constraints, and exhibits superior recovery performance, particularly in nonconvex contexts. While the standard PG method [6] ensures a monotonically decreasing function with a constant step-size, it is often hampered by slow convergence due to step-sizes being inversely proportional to signal size. Our method overcomes this through a modified PG step featuring an optimized, signal size-independent step-size. Additionally, we introduce an innovative low-rank matrix subspace projection step, significantly enhancing recovery in

scenarios with low sampling rates and moderate SSS orders. We demonstrate the efficacy of both steps in achieving guaranteed convergence.

3. **Efficient Implementations:** To enhance the efficiency of our two-step iterative method, we have implemented strategies to reduce both the number of iterations and the computational complexity of each iteration. By fully exploiting the Hankel and low-rank structures, we have reduced computational demands and storage needs for large-sized variables in the iterative process. Furthermore, we have streamlined the optimization by converting a dual-variable problem into a single-variable one, utilizing subgradient conditions. This optimization reduces computational complexity from $\mathcal{O}(n^3)$ to $\mathcal{O}(r^4 n + r^3 n \log n)$ per iteration.

In the LPPG method, the Hankel enforcement parameter holds an inverse proportional relationship with the step-size in the gradient descent process. This parameter effectively establishes a strict upper limit for the step-size, thereby creating a direct link between the noise level and the convergence speed. Unlike other descent methods that necessitate complex estimations or line-search algorithms to determine the step-size based on initial conditions, the LPPG method offers an efficient way to set this parameter once the noise level is ascertained. For example, in noiseless scenarios, it is recommended to set the parameter for structural constraints close to zero, as outlined in [88]. This recommendation implies that a substantially larger step-size can be chosen for rapid convergence when the Hankel enforcement parameter is minimal. Furthermore, the LPPG approach paves the way for applying fast variations of the PG method [106, 116] in the realm of SCS.

To substantiate the effectiveness of our proposed LPPG method, extensive numerical simulations have been conducted. The outcomes from these simulations attest to the superiority of the LPPG method over existing benchmarks in four key aspects: 1) It requires fewer samples for accurate estimation of SSS with a higher model order; 2) It achieves faster convergence in scenarios devoid of noise; 3) It retains robust performance even in the presence of significant noise; 4) It enhances accuracy by effectively mitigating the biased weighted norms associated with SSS. These advantages underscore the LPPG method's potential in efficiently and accurately

addressing challenges in SCS.

4.2 The Optimization Formulation

We consider an order-r SSS[1] $\boldsymbol{x} \in \mathbb{C}^n$ (with $r \ll n$), representing a superposition of complex sinusoids:

$$\boldsymbol{x} = \sum_{k=1}^{r} d_k \boldsymbol{y}(f_k, \tau_k;\ n), \tag{4.1}$$

where $d_k \in \mathbb{C}$ is the complex amplitude, and $\boldsymbol{y}(f_k, \tau_k;\ n) \coloneqq [1, e^{i2\pi f_k - \tau_k}, \ldots, e^{(i2\pi f_k - \tau_k)(n-1)}]^\mathsf{T} \in \mathbb{C}^n$ represents the kth spectral component with normalized frequency $f_k \in [0, 1)$ and damping factor τ_k [56]. Due to practical constraints, often only a subset of $\boldsymbol{x}$ is accessible for observation. Denote $\Omega \subset \{1, \ldots, n\}$ as the set of indices corresponding to the observed entries, with $|\Omega| \coloneqq m < n$, and let $\mathcal{P}_\Omega$ be the sampling operator that captures entries indexed by Ω while zeroing out the others. The sampling rate is thus defined as $Sp \coloneqq m/n < 1$.

The primary challenge lies in reconstructing the SSS $\boldsymbol{x}$ from its partial observations $\boldsymbol{s}$, i.e.,

$$\text{find} \quad \boldsymbol{x} \quad \text{s.t.} \quad \mathcal{P}_\Omega \boldsymbol{x} = \boldsymbol{s}. \tag{4.2}$$

The Hankel matrix associated with $\boldsymbol{x}$, denoted as $\mathcal{H}\boldsymbol{x}$, admits a Vandermonde decomposition:

$$\mathcal{H}\boldsymbol{x} = \sum_{k=1}^{r} d_k \boldsymbol{y}(f_k, \tau_k;\ p)\boldsymbol{y}(f_k, \tau_k;\ q)^\mathsf{T} \in \mathbb{C}^{p \times q}. \tag{4.3}$$

Typically, p and q are chosen such that $p \approx q \approx n/2$, resulting in $\mathcal{H}\boldsymbol{x}$ being approximately square [2, 56]. It is evident that $\operatorname{rank}(\mathcal{H}\boldsymbol{x}) \leq r \ll q$ [31], implying a low-rank structure that is central to our recovery approach.

Consequently, the recovery of SSS can be recast as a problem of low-rank Hankel matrix recovery [2, 53, 55–57]. To capture the data discrepancy effectively while preserving the Hankel

[1]For simplicity, our discussion is centered on one-dimensional SSS in this chapter. However, the proposed method is adaptable to multilevel cases.

structure, we introduce an enhanced Hankel matrix $\boldsymbol{X}^e = \mathcal{H}\boldsymbol{x}$. The rank-constrained nonconvex optimization problem is then formulated as follows:

$$\min_{\boldsymbol{X}^e,\boldsymbol{x}} \ \delta(\text{rank}(\boldsymbol{X}^e) \leq r) + \frac{1}{2}\|\boldsymbol{s} - \mathcal{P}_\Omega \boldsymbol{x}\|_2^2 \quad \text{s.t. } \boldsymbol{X}^e = \mathcal{H}\boldsymbol{x}, \tag{4.4}$$

where $\delta(\cdot)$ denotes the indicator function for rank-constrained matrices, defined as zero when its argument is true, and infinity otherwise:

$$\delta(x) = \begin{cases} 0 & x = 1 \\ +\infty & x = 0. \end{cases} \tag{4.5}$$

To circumvent the difficulties inherent in solving this constrained nonconvex optimization problem, we adopt a relaxation approach:

$$\min_{\boldsymbol{X}^e,\boldsymbol{x}} \ \delta(\text{rank}(\boldsymbol{X}^e) \leq r) + \frac{1}{2}\|\boldsymbol{s} - \mathcal{P}_\Omega \boldsymbol{x}\|^2 + \frac{\beta}{2}\|\boldsymbol{X}^e - \mathcal{H}\boldsymbol{x}\|_F^2 + \frac{\alpha}{2}\|\boldsymbol{x}\|^2, \tag{4.6}$$

where $\beta > 0$ serves as the Hankel enforcement parameter, ensuring adherence to the Hankel structure, and α is a small positive scalar introduced for regularization purposes. While the PG algorithm [6] can solve problem (4.6) with a convergence guarantee, its direct application often results in slow convergence. This sluggishness is attributed to the unequal weighted norm [63,64] and the high computational complexity per iteration of $\mathcal{O}(n^3)$, primarily driven by the SVD operation required by the rank constraint.

Remark 6. *For ease of exposition and to maintain focus on the core principles, we limit our discussion to one-dimensional SSS in Sections 4.2 and 4.3. However, we extend our method to higher-dimensional cases and present the corresponding simulation results in Section 4.5.*

We highlight the key advantages of our formulation as follows:

1. **Hankel Enforcement Parameter (β):** Our formulation uniquely incorporates the parameter β, allowing for adjustments based on different noise levels. This yields two

significant benefits:

(a) In low noise scenarios, a smaller value of β can be selected [88], facilitating faster convergence as the step-size in gradient-based methods is inversely proportional to β [56, 57].

(b) For high noise levels, a larger β ensures more accurate reconstruction by aligning more closely with the original constrained optimization, especially when data fidelity is unreliable.

2. **Data Fidelity ($\|\boldsymbol{s}-\mathcal{P}_\Omega \boldsymbol{x}\|^2$):** Our chosen data fidelity term contrasts with $\|\mathcal{H}(\boldsymbol{s}-\mathcal{P}_\Omega \boldsymbol{x})\|_F^2$ used in previous works [2, 14, 56–58], where varying weighting coefficients for individual entries could introduce bias in the solution. Our approach ensures equal treatment of noise terms, preventing potential biases due to unequal contributions.

The efficacy of our formulation is demonstrated through numerical simulations, as shown in Section 4.5.

4.3 A Low-rank Projected Proximal Gradient Method

The formulation we propose represents a nonconvex, unconstrained optimization problem, which distinctively includes nonsmooth terms. Our main objective is to achieve the critical points $(\boldsymbol{x}^\star, \boldsymbol{X}^{e\star})$ of the objective function, as delineated in (4.6). This task is carried out with respect to the observed samples, aiming to accurately recover the underlying spectral components represented by these critical points.

4.3.1 The Modified Proximal Gradient Step

Given the proximal operator defined as:

$$\operatorname{prox}_{\gamma g}(\boldsymbol{v}) = \arg\ \min_{\boldsymbol{x}} \left(g(\boldsymbol{x}) + \frac{1}{2\gamma}\|\boldsymbol{x}-\boldsymbol{v}\|^2 \right). \tag{4.7}$$

For a function g, if prox_g has a closed or semi-closed form solution and is computationally easy to evaluate [6, 97], we generally describe such functions as proximable [117]. Let $f : \mathbb{R}^d \to \mathbb{R}$ be a function. We say that f is Lipschitz continuous if there exists a constant $L_f \geq 0$ such that for all $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^d$,

$$\|f(x) - f(y)\|_2 \leq L_f \|x - y\|_2.$$

The constant L_f is called the Lipschitz constant of f. The standard PG method process is detailed in Algorithm 5 in [6]. It addresses unconstrained optimization problems formulated as:

$$\min_{\boldsymbol{x}} \; F(\boldsymbol{x}) \coloneqq \min_{\boldsymbol{x}} \; f(\boldsymbol{x}) + g(\boldsymbol{x}), \tag{4.8}$$

where $f(\cdot)$ is a Lipschitz differentiable function[2], and $g(\cdot)$ is usually a proximable function.

It is crucial to note that the step size $0 < \gamma \leq 1/L_{\nabla f}$ is selected inversely proportional to the Lipschitz constant $L_{\nabla f}$.

Algorithm 5 The Standard PG Method [6]

Initialize $\boldsymbol{x}_0$ and select $0 < \gamma \leq \frac{1}{L_{\nabla f}}$ (where $L_{\nabla f}$ is the Lipschitz constant of ∇f).
for $k = 0, 1, \ldots$ **do**
  Update $\boldsymbol{x}_{k+1} \leftarrow \text{prox}_{\gamma g}\left(\boldsymbol{x}_k - \gamma \nabla f(\boldsymbol{x}_k)\right)$.
end for

The PG method, as described above, is effective in converging within nonconvex and nonsmooth optimization frameworks, especially when the step size is carefully selected. This gradient-descent based methodology is notable for consistently reducing the objective function and demonstrating enhanced robustness, particularly in comparison to alternating projection methods that require random initialization conditions [56].

When the PG method is applied directly to the function formulated in (4.6), we define:

$$f(\boldsymbol{X}^e, \boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{s} - \mathcal{P}_\Omega \boldsymbol{x}\|^2 + \frac{\beta}{2}\|\boldsymbol{X}^e - \mathcal{H}\boldsymbol{x}\|_F^2 + \frac{\alpha}{2}\|\boldsymbol{x}\|^2. \tag{4.9}$$

The Lipschitz constant for the gradient of f, denoted as $L_{\nabla f}$, is determined to be $1 + \beta \min(p, q) +$

[2]The gradient ∇f is Lipschitz continuous, with a Lipschitz constant denoted as $L_{\nabla f}$.

α. Here, p and q are dimensions pertaining to the Hankel mapping as indicated in (2.1). As a result, the step-size selected for the PG method becomes inversely proportional to the size of the signal. This relationship unfortunately results in excessively slow convergence, especially for signals of moderate or large size, rendering this direct application of the PG method impractical for such scenarios.

In our methodology, we leverage the necessary condition for critical points, which implies that the partial derivative with respect to $\boldsymbol{x}$ must be zero. This implies that minimizing $\boldsymbol{x}$ is a fundamental step. It is important to note that the variables $\boldsymbol{x}$ and $\boldsymbol{X}^e$ are linearly coupled solely within the term $\|\boldsymbol{X}^e - \mathcal{H}\boldsymbol{x}\|_F^2$. This observation leads us to adopt a substitution strategy to deduce the critical points $\boldsymbol{x}^\star$ in terms of $\boldsymbol{X}^e$. Such a substitution is pivotal for reconceptualizing the objective function to encompass only the single variable $\boldsymbol{X}^e$:

$$F(\boldsymbol{X}^e) \coloneqq f(\boldsymbol{X}^e) + g(\boldsymbol{X}^e), \tag{4.10}$$

$$f(\boldsymbol{X}^e) \coloneqq \min_{\boldsymbol{x}} \ \frac{1}{2}\|\boldsymbol{s} - \mathcal{P}_\Omega \boldsymbol{x}\|^2 + \frac{\beta}{2}\|\boldsymbol{X}^e - \mathcal{H}\boldsymbol{x}\|_F^2 + \frac{\alpha}{2}\|\boldsymbol{x}\|^2, \tag{4.11}$$

$$g(\boldsymbol{X}^e) \coloneqq \delta(\operatorname{rank}(\boldsymbol{X}^e) \leq r). \tag{4.12}$$

Because of the lifting technique for $\boldsymbol{X}^e$, the function $g(\boldsymbol{X}^e)$ is proximable. Moreover, the lifting variable $\boldsymbol{X}^e$ also makes the Hankel structure constraint more flexible according to the scale of Hankel enforcement parameter β. This reformulation not only simplifies the optimization problem but also aligns it more closely with the underlying structure of our Hankel matrix-based approach in error estimation.

Upon solving for the gradient with respect to $\boldsymbol{x}$, denoted as $\nabla_{\boldsymbol{x}}$, we derive the optimal $\boldsymbol{x}^\star$:

$$\boldsymbol{x}^\star = \mathcal{L}(\mathcal{P}_\Omega^* \boldsymbol{s} + \beta \mathcal{H}^* \boldsymbol{X}^e), \quad \mathcal{L} = (\alpha \mathcal{I} + \mathcal{P}_\Omega^* \mathcal{P}_\Omega + \beta \mathcal{H}^* \mathcal{H})^{-1}. \tag{4.13}$$

Remark 7. *The operator $\mathcal{L} = (\alpha \mathcal{I} + \mathcal{P}_\Omega^* \mathcal{P}_\Omega + \beta \mathcal{H}^* \mathcal{H})^{-1}$ is guaranteed to exist and can be represented by diagonal matrices with positive entries. This assurance stems from the fact that*

$\mathcal{H}^*\mathcal{H}$ scales each element of an n-dimensional vector by w_a, the count of elements in the ath skew-diagonal of a $p \times q$ matrix. Additionally, $\mathcal{P}_\Omega^*\mathcal{P}_\Omega$, which can be equated to $\mathcal{P}_\Omega$, is represented by diagonal matrices where the observed entries are marked with 1s, and the unobserved entries are filled with 0s.

A notable aspect of our formulation in (4.10) is the bounding of the Lipschitz constant of $\nabla f(\boldsymbol{X}^e)$ by β, which is remarkably independent of the dimensionality of the problem. This characteristic facilitates the selection of a larger step-size for the corresponding PG method, thereby accelerating convergence in large-scale scenarios. The Lipschitz constant of $\nabla f(\boldsymbol{X}^e)$, denoted as $L_{\nabla f}$, is calculated in Proposition 1:

Proposition 1. *The Lipschitz constant of $\nabla f(\boldsymbol{X}^e)$, denoted as $L_{\nabla f}$. We have*

$$\nabla f(\boldsymbol{X}^e) = -\beta\mathcal{H}\mathcal{L}\mathcal{P}_\Omega^*\boldsymbol{s} - \beta^2\mathcal{H}\mathcal{L}\mathcal{H}^*\boldsymbol{X}^e + \beta\boldsymbol{X}^e, \tag{4.14}$$

and hence $L_{\nabla f} < \beta$.

Remark 8. *This reduced upper bound on the Lipschitz constant, L_f, allows for an increased step-size, $\gamma = 1/\beta$, in iterative PG algorithms. This optimized step-size is crucial in achieving efficient convergence.*

Efficiency in computations is a key strength of our proximal gradient step. It primarily entails the evaluation of $\nabla f(\boldsymbol{X}^e)$ as expressed in (4.14), in addition to computing the proximal operator:

$$\operatorname{prox}_{\gamma g}\left(\boldsymbol{X}_k^e - \gamma\nabla f(\boldsymbol{X}_k^e)\right) = \mathcal{T}_r\left(\boldsymbol{X}_k^e - \gamma\nabla f(\boldsymbol{X}_k^e)\right), \tag{4.15}$$

where $\mathcal{T}_r(\cdot)$ represents the truncated SVD operator [118]. Importantly, $\mathcal{T}_r$ efficiently handles the combination of a Hankel matrix and a low-rank matrix, employing the Lanczos method to calculate SVD's extreme triplets via matrix-vector multiplications. Furthermore, Hankel matrix-vector multiplications can be accelerated from $\mathcal{O}(n^2)$ to $\mathcal{O}(n\log n)$ using the FFT [3], significantly enhancing computational speed.

4.3.2 The Low-Rank Projection Step

In the realm of Hankel low-rank matrix completion, the number of stationary points is known to scale exponentially with the rank r and linearly with the size n [14]. A pivotal challenge in this context is the selection of optimal critical points that lead to more accurate signal reconstructions. Common practices, such as those employed in Cadzow's algorithm, typically involve a straightforward and cost-effective scalar correction to minimize the cost function while maintaining the matrix rank [64].

Motivated by this approach, we have incorporated a subspace projection step into our methodology. This step is designed to create a new low-rank matrix $\boldsymbol{X}^e_{k,\frac{1}{2}}$, which is intended to more closely align with the Hankel structure and the observed samples. This alignment is achieved by leveraging the output from the k-th iteration of the proximal gradient step, denoted as $\boldsymbol{X}^e_k$. The primary goal of this subspace projection step is to refine the approximation of $\boldsymbol{X}^e_k$ such that it better represents the underlying Hankel structure and sample data, thereby enhancing the accuracy of the overall signal recovery process.

Specifically, the rank-r matrix $\boldsymbol{X}^e_k$ undergoes preprocessing to achieve a reduced Singular Value Decomposition (rSVD): $\boldsymbol{X}^e_k = \boldsymbol{U}_k\boldsymbol{\Sigma}_k\boldsymbol{V}_k^{\mathsf{H}}$, where $\boldsymbol{U}_k \in \mathbb{C}^{p\times r}$ and $\boldsymbol{V}_k \in \mathbb{C}^{q\times r}$ consist of orthonormal columns. Maintaining $\boldsymbol{U}_k$ and $\boldsymbol{V}_k$ as fixed, the optimization problem in (4.6) is redefined as:

$$\begin{aligned}\min_{\boldsymbol{X}^e_k,\boldsymbol{x}} &\delta(\operatorname{rank}(\boldsymbol{X}^e_k) \le r) + \frac{1}{2}\|\boldsymbol{s} - \mathcal{P}_\Omega\boldsymbol{x}\|^2 + \frac{\beta}{2}\|\boldsymbol{X}^e_k - \mathcal{H}\boldsymbol{x}\|_F^2 + \frac{\alpha}{2}\|\boldsymbol{x}\|^2 \\ &= \min_{\boldsymbol{\Sigma},\boldsymbol{x}} \ \frac{1}{2}\|\boldsymbol{s} - \mathcal{P}_\Omega\boldsymbol{x}\|^2 + \frac{\beta}{2}\|\boldsymbol{U}_k\boldsymbol{\Sigma}\boldsymbol{V}_k^{\mathsf{H}} - \mathcal{H}\boldsymbol{x}\|_F^2 + \frac{\alpha}{2}\|\boldsymbol{x}\|^2, \end{aligned} \tag{4.16}$$

resulting in a least-squares problem involving the variables $\boldsymbol{\Sigma}$ and $\boldsymbol{x}$, for which there exists a unique closed-form solution[3]. The low-rank projection in (4.16) is equivalent to solving the

[3]Note that $\boldsymbol{\Sigma}$ is not assumed to be diagonal.

following system of linear equations:

$$\underbrace{\begin{bmatrix} \mathcal{P}_\Omega + \beta\mathcal{W} & -\beta\mathcal{H}^*\mathcal{P}_{U,V} \\ -\beta\mathcal{P}_{U,V}^*\mathcal{H} & \beta\mathcal{P}_{U,V}^*\mathcal{P}_{U,V} \end{bmatrix}}_{\mathcal{A}} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{\Sigma} \end{bmatrix} = \begin{bmatrix} \mathcal{P}_\Omega \boldsymbol{s} \\ \mathbf{0} \end{bmatrix}, \tag{4.17}$$

where $\mathcal{P}_{U,V}\boldsymbol{\Sigma} = \boldsymbol{U}_k\boldsymbol{\Sigma}\boldsymbol{V}_k^{\mathsf{H}}$ and $\mathcal{P}_{U,V}^*\boldsymbol{X}^e = \boldsymbol{U}_k^{\mathsf{H}}\boldsymbol{X}^e\boldsymbol{V}_k$. The equation (4.17) can be resolved using conjugate gradient methods [51], typically featuring a computational complexity of $\mathcal{O}((n+r^2)^3)$. However, our preliminary simulations suggest that this level of complexity may be excessive for larger scale problems with a substantial value of n.

As previously demonstrated, we can recast the optimization problem in (4.16) into a single-variable formulation (4.18), thereby considerably reducing the computational complexity from $\mathcal{O}((n+r^2)^3)$ to $\mathcal{O}(r^3 n\log n + r^4 n)$. Specifically, we reshape (4.16) as follows:

$$\begin{aligned} &\min_{\boldsymbol{\Sigma}_k} \; L(\boldsymbol{\Sigma}_k), \text{ with} \\ &L(\boldsymbol{\Sigma}_k) = \min_{\boldsymbol{x}} \frac{1}{2}\|\boldsymbol{s} - \mathcal{P}_\Omega\boldsymbol{x}\|^2 + \frac{\beta}{2}\|\boldsymbol{U}_k\boldsymbol{\Sigma}_k\boldsymbol{V}_k^{\mathsf{H}} - \mathcal{H}\boldsymbol{x}\|_F^2 + \frac{\alpha}{2}\|\boldsymbol{x}\|^2. \end{aligned} \tag{4.18}$$

In this new formulation, the objective function $L(\boldsymbol{\Sigma}_k)$ solely depends on $\boldsymbol{\Sigma}_k$, though its computation also involves optimizing $\boldsymbol{x}$. The optimal solutions to (4.18) are outlined below.

Proposition 2. *Define* $\mathcal{P}_{\boldsymbol{U}_k,\boldsymbol{V}_k}\boldsymbol{\Sigma} = \boldsymbol{U}_k\boldsymbol{\Sigma}\boldsymbol{V}_k^{\mathsf{H}}$ *and* $\mathcal{P}_{\boldsymbol{U}_k,\boldsymbol{V}_k}^*\boldsymbol{X}^e = \boldsymbol{U}_k^{\mathsf{H}}\boldsymbol{X}^e\boldsymbol{V}_k$. *The optimal solutions to* (4.18) *and* (4.16) *are given by:*

$$\boldsymbol{x}^\dagger = \left(\alpha\mathcal{I} + \mathcal{P}_\Omega^*\mathcal{P}_\Omega + \beta\mathcal{H}^*\mathcal{H}\right)^{-1}\left(\mathcal{P}_\Omega^*\boldsymbol{s} + \beta\mathcal{H}^*\mathcal{P}_{\boldsymbol{U}_k,\boldsymbol{V}_k}\boldsymbol{\Sigma}_k^\star\right), \tag{4.19}$$

$$\boldsymbol{\Sigma}_k^\star = \left(\mathcal{I} - \beta\mathcal{P}_{\boldsymbol{U}_k,\boldsymbol{V}_k}^*\mathcal{H}\mathcal{L}\mathcal{H}^*\mathcal{P}_{\boldsymbol{U}_k,\boldsymbol{V}_k}\right)^{-1}\left(\mathcal{P}_{\boldsymbol{U}_k,\boldsymbol{V}_k}^*\mathcal{H}\mathcal{L}\mathcal{P}_\Omega^*\boldsymbol{s}\right). \tag{4.20}$$

Remark 9. *The regularization term* $\frac{\alpha}{2}\|\boldsymbol{x}\|^2$ *ensures that the matrix* $\mathcal{I} - \beta\mathcal{P}_{\boldsymbol{U}_k,\boldsymbol{V}_k}^*\mathcal{H}\mathcal{L}\mathcal{H}^*\mathcal{P}_{\boldsymbol{U}_k,\boldsymbol{V}_k}$ *is positive definite, guaranteeing the existence of its inverse linear mapping.*

Proposition 3. *Let* $\boldsymbol{X}_{k,\frac{1}{2}}^e$ *be defined as* $\boldsymbol{U}_k\boldsymbol{\Sigma}_k^\star\boldsymbol{V}_k^{\mathsf{H}}$. *The low-rank projection step exhibits the*

sufficient decrease property:

$$F(\boldsymbol{X}^e_{k,\frac{1}{2}}) \leq F(\boldsymbol{X}^e_k) - \left(\frac{\alpha}{\alpha+\beta}\right) \|\boldsymbol{X}^e_{k,\frac{1}{2}} - \boldsymbol{X}^e_k\|_F^2. \tag{4.21}$$

The computation of (4.19) and (4.20) benefits greatly from the Hankel and low-rank structure of the involved matrices. The reduction in the number of variables from $n + r^2$ to r^2 considerably lowers the computational complexity. This makes it feasible to determine the optimal values using the conjugate gradient (CG) method for linear mapping. The computational complexity for this low-rank projection step is $\mathcal{O}(r^3 n \log n + r^4 n)$. A detailed analysis of this complexity is provided in Section 4.4.

This step in the algorithm can be viewed as a refinement of the SVD process, aiming to enhance data fidelity while preserving the Hankel structure after each MPG step. It serves as a critical adjustment, ensuring that each iteration of the algorithm progresses towards a more accurate reconstruction of the SSS.

4.3.3 The LPPG Algorithm with Convergence

We incorporate the MPG step with an optimized step-size and the low-rank subspace projection into our LPPG algorithm. This integration is articulated in Algorithm 6.

Algorithm 6 LPPG for SCS

Input: $\boldsymbol{s}$, β, α, ϵ, $\gamma = \frac{1}{\beta}$, initial guess $\boldsymbol{X}^e_0 \leftarrow \mathcal{T}_r \mathcal{H}\boldsymbol{s}$.

while $\|\partial F(\boldsymbol{X}^e_k)\|_F \geq \epsilon \|\boldsymbol{X}^e_k\|_F$ **do**

Compute $\boldsymbol{X}^e_{k,\frac{1}{2}} \leftarrow \boldsymbol{U}_k \boldsymbol{\Sigma}^\star{}_k \boldsymbol{V}_k^{\mathsf{H}}$ through subspace projection of $\boldsymbol{X}^e_k$.

Update $\boldsymbol{X}^e_{k+1} \leftarrow \mathcal{T}_r \left(\boldsymbol{X}^e_{k,\frac{1}{2}} - \gamma \nabla f(\boldsymbol{X}^e_{k,\frac{1}{2}})\right)$.

end while

Output: The reconstructed signal $\boldsymbol{X}^e_k$ from the last iteration, and $\boldsymbol{x}^\dagger \leftarrow \mathcal{L}(\mathcal{P}^*_\Omega \boldsymbol{s} + \beta \mathcal{H}^* \boldsymbol{X}^e_k)$.

The objective function (4.10) consistently exhibits a monotonic decrease after each iteration, owing to the meticulous optimization of the subproblem. The convergence properties of the LPPG algorithm are further detailed in Theorem 1.

Theorem 1. *The sequences* $\{\boldsymbol{X}^e_{k,\frac{1}{2}}\}$ *and* $\{\boldsymbol{X}^e_k\}$ *generated by Algorithm 6 are bounded. Suppose* $\boldsymbol{X}^{e*}$ *is an accumulation point of the sequence* $\{\boldsymbol{X}^e_k\}$*. In that case, it satisfies* $0 \in \partial F(\boldsymbol{X}^{e*})$*, meaning* $\boldsymbol{X}^{e*}$ *is a critical point of the objective function. Furthermore, the following properties hold:*

$$-\nabla f(\boldsymbol{X}^e_{k,\frac{1}{2}}) + \nabla f(\boldsymbol{X}^e_{k+1}) - \frac{1}{\gamma}(\boldsymbol{X}^e_{k+1} - \boldsymbol{X}^e_{k,\frac{1}{2}}) \in \partial F(\boldsymbol{X}^e_{k+1}), \tag{4.22}$$

and

$$\| - \nabla f(\boldsymbol{X}^e_{k,\frac{1}{2}}) + \nabla f(\boldsymbol{X}^e_{k+1}) - \frac{1}{\gamma}(\boldsymbol{X}^e_{k+1} - \boldsymbol{X}^e_{k,\frac{1}{2}})\|_F \tag{4.23}$$

$$\leq \left(\frac{1}{\gamma} + L_{\nabla f}\right) \|\boldsymbol{X}^e_{k+1} - \boldsymbol{X}^e_{k,\frac{1}{2}}\|_F \to 0 \quad as \quad k \to \infty. \tag{4.24}$$

The convergence rate of the algorithm is characterized by:

$$\min_{i=0,\cdots,K} \|\partial F(\boldsymbol{X}^e_{i+1})\|_F^2 \leq \frac{c_0}{K+1}(F(\boldsymbol{X}^e_0) - F^*),$$

where $c_0 = \frac{(\beta - L_{\nabla f})\beta^4}{(\alpha+\beta)^2}$.

The proof of Theorem 1 encompasses three key steps: 1) Establishing a sufficient decrease in the objective function; 2) Proving that any accumulation point $\boldsymbol{X}^{e*}$ qualifies as a critical point of the function; and 3) Determining the convergence rate of the algorithm. A detailed exposition of the proof is provided in Section 4.7.4.

Remark 10. *The crux of the convergence in the LPPG algorithm stems from the MPG step. This step ensures that* $\|\partial F\|_F \leq \left(\frac{1}{\gamma} + L_{\nabla f}\right) \|\boldsymbol{X}^e_{k+1} - \boldsymbol{X}^e_{k,\frac{1}{2}}\|_F$*, which diminishes as* k *increases. Additionally, the inclusion of the low-rank subspace projection step contributes to a further decrease in the objective function. This leads to an accelerated convergence and empirically yields better signal reconstruction. This enhancement is primarily due to the better alignment with the observed samples and the intrinsic Hankel structure, as demonstrated in our numerical experiments.*

4.3.4 Stopping Criteria

Previous research on non-convex methods, such as those cited in [54, 56–58, 119], commonly employed stopping criteria based on the relative difference between successive iterations, quantified as $\|\boldsymbol{X}^e_{k+1} - \boldsymbol{X}^e_k\|_2 / \|\boldsymbol{X}^e_k\|_2$. This criterion tends to be effective when the initial approximation is close to the global minimum's basin of attraction. However, this may not always be sufficient in practical scenarios, especially when the frequencies to be estimated are closely spaced, as small step sizes might inadvertently lead the algorithm away from the true solution.

To enhance robustness, our LPPG method directly assesses convergence to critical points through the subgradient of F. In verifying the accumulation point of $\{\boldsymbol{X}^e_k\}$, we ensure that $\partial F(\boldsymbol{X}^e_k)$ tends towards zero. Given the non-differentiability of g, we utilize the optimality condition of the proximal operator (4.15) to infer:

$$-\frac{1}{\gamma}(\boldsymbol{X}^e_{k+1} - (\boldsymbol{X}^e_{k,\frac{1}{2}} - \gamma \nabla f(\boldsymbol{X}^e_{k,\frac{1}{2}}))) \in \partial g(\boldsymbol{X}^e_{k+1}). \tag{4.25}$$

This leads us to the following formulation:

$$\begin{aligned}
&-\frac{1}{\gamma}(\boldsymbol{X}^e_{k+1} - (\boldsymbol{X}^e_{k,\frac{1}{2}} - \gamma \nabla f(\boldsymbol{X}^e_{k,\frac{1}{2}}))) + \nabla f(\boldsymbol{X}^e_{k+1}) \\
=&\frac{1}{\gamma}(\boldsymbol{X}^e_{k,\frac{1}{2}} - \boldsymbol{X}^e_{k+1}) + (\nabla f(\boldsymbol{X}^e_{k+1}) - \nabla f(\boldsymbol{X}^e_{k,\frac{1}{2}})) \\
=&((\frac{1}{\gamma} - \beta)\mathcal{I} + \beta^2 \mathcal{H}\mathcal{L}\mathcal{H}^*)(\boldsymbol{X}^e_{k,\frac{1}{2}} - \boldsymbol{X}^e_{k+1}) \\
=&\beta^2 \mathcal{H}\mathcal{L}\mathcal{H}^*(\boldsymbol{X}^e_{k,\frac{1}{2}} - \boldsymbol{X}^e_{k+1}) \in \partial F(\boldsymbol{X}^e_{k+1}).
\end{aligned} \tag{4.26}$$

This equation facilitates an efficient check for $\beta^2 \mathcal{H}\mathcal{L}\mathcal{H}^*(\boldsymbol{X}^e_{k,\frac{1}{2}} - \boldsymbol{X}^e_{k+1})$, as detailed in Section 4.4.

The choice of β is pivotal for the stopping criteria: in noiseless scenarios, large steps (small β) are expected, driving the subgradient of F towards zero. Conversely, in heavily noisy conditions (large β), the subgradient of F might oscillate near local minima, suggesting an earlier termination. Moreover, once β is fixed, the difference in the MPG step can be used to

measure the subgradient of F, as indicated in (4.26). The LPPG method with this stopping criteria proves more robust than the traditional relative difference between iterations, as it remains unaffected by the implicit step size.

4.4 Computational Complexity Analysis

In this section, we emphasize the significant reduction in computational complexity achieved with our proposed LPPG algorithm. Each iteration of the LPPG algorithm incurs a computational complexity of $\mathcal{O}(r^4 n + r^3 n \log n)$, which is markedly lower than the $\mathcal{O}(n^3)$ complexity typically encountered in large-size SSS signal processing. This substantial efficiency gain stems from the strategic exploitation of the low-rank and Hankel structures inherent to the problem. As a result, our LPPG algorithm stands out as a highly effective tool for tackling larger scale SSS recovery tasks, offering a notable advancement in computational efficiency.

We base our computational complexity analysis on several fundamental operations relevant to our algorithm:

The computational complexity analysis of our proposed LPPG algorithm is based on several key operations:

- **Hankel Matrix-vector Multiplication:** Efficiently performed using the FFT for a $p \times q$ Hankel matrix mapped from an array of size n, this operation requires $\mathcal{O}(n \log n)$ flops [51].

- **Low-Rank Matrix-vector Multiplication:** The multiplication of a rank-r, $n \times n$ square matrix with a vector involves $\mathcal{O}(rn)$ operations. Given a rank r matrix $\boldsymbol{A} \in \mathbb{C}^{n \times n}$ in SVD form ($\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathsf{H}}$), the complexity of multiplying $\boldsymbol{A}$ by a vector $\boldsymbol{x}$ can be broken down as follows:

$$\underbrace{\boldsymbol{A}}_{n\times n} \times \underbrace{\boldsymbol{x}}_{n\times 1} = \underbrace{\boldsymbol{U}}_{n\times r} \times [\underbrace{\boldsymbol{\Sigma}}_{r\times r} \times (\underbrace{\boldsymbol{V}^{\mathsf{H}}}_{r\times n} \times \underbrace{\boldsymbol{x}}_{n\times 1})] \tag{4.27}$$

  Here, computing $\boldsymbol{V}^{\mathsf{H}}\boldsymbol{x}$ incurs $\mathcal{O}(rn)$ flops, followed by $\mathcal{O}(r^2)$ for multiplying $\boldsymbol{\Sigma}$ and $\mathcal{O}(rn)$

for the final product with $\boldsymbol{U}$ (considering $r \leq n$).

- **Hankel Mapping $\mathcal{H}$:** Forming a $p \times q$ Hankel matrix $\mathcal{H}x$ from $\boldsymbol{x}$ of size n is done with $\mathcal{O}(n)$ complexity, leveraging the repeated elements in $\mathcal{H}x$. Operations like addition, scaling, and inner products within the Hankel subspace require $\mathcal{O}(n)$ flops.

- **Adjoint of Hankel Mapping $\mathcal{H}^*$:** The transformation from $p \times q$ matrices to n-dimensional vectors via $\mathcal{H}^*$ is optimized in the LPPG algorithm. Given that $\{\boldsymbol{X}_k^e\}$ and $\{\boldsymbol{X}_{k,\frac{1}{2}}^e\}$ are rank-r matrices, they can be decomposed as $\boldsymbol{X}^e = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathsf{H}}$. In our method, $\mathcal{H}^*$ operates post-$\mathcal{P}_{\boldsymbol{U},\boldsymbol{V}}$ as follows:

$$\begin{aligned} &\mathcal{H}^*\mathcal{P}_{\boldsymbol{U},\boldsymbol{V}}\boldsymbol{\Sigma} = \mathcal{H}^*(\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathsf{H}}) \\ =&\sum_{i=1}^{r} \Sigma(i,i)\mathcal{H}^*[\boldsymbol{U}(:,i)\boldsymbol{V}(:,i)^{\mathsf{H}}]. \end{aligned} \tag{4.28}$$

  The computational complexity of $\mathcal{H}^*\mathcal{P}_{\boldsymbol{U},\boldsymbol{V}}\boldsymbol{\Sigma}$ is $\mathcal{O}(r^2n + rn\log n)$ flops, considering the use of fast convolution techniques.

With the aforementioned preliminaries, it is evident that the LPPG algorithm effectively improves its computational efficiency with to handle larger scale SSS recovery problems, leveraging the unique structures of the matrices involved. Building on these efficient calculations, we analyze the computational complexity in each iteration, which encompasses the following three parts:

- **Modified PG Step:** As delineated in the proximal mapping (4.15), the modified PG step entails solving a rank-r truncated singular value decomposition (tSVD) of a linear combination of a Hankel matrix and a low-rank matrix. Constructing this Hankel matrix demands $\mathcal{O}(r^2n + rn\log n)$ flops. We implement the fast r-truncated SVD for the linear combination matrix using the Lanczos method [5,120] with $\mathcal{O}(rn\log n + r^2n)$ flops, based on the fast matrix-vector multiplication of the Hankel matrix and low-rank matrix. Therefore, the total computational complexity of the modified PG step is $\mathcal{O}(r^2n + rn\log n)$.

- **Subspace Projection Step:** This step involves solving the following equation for $\boldsymbol{\Sigma}$:

$$\left(\mathcal{I} - \beta \mathcal{P}^*_{\boldsymbol{U},\boldsymbol{V}} \mathcal{H}\mathcal{L}\mathcal{H}^* \mathcal{P}_{\boldsymbol{U},\boldsymbol{V}}\right) \boldsymbol{\Sigma} = \mathcal{P}^*_{\boldsymbol{U},\boldsymbol{V}} \mathcal{H}\mathcal{L}\mathcal{P}^*_{\Omega} \boldsymbol{s}. \tag{4.29}$$

  This general linear equation can be solved using the conjugate gradient method [121] with at most r^2 iterations, where r^2 is the size of the variable $\mathrm{vec}(\boldsymbol{\Sigma})$. We have shown that each iteration of the linear mapping in (4.29) requires $\mathcal{O}(nr^2 + rn\log n)$ floating-point operations (flops). Hence, the total computational complexity for the subspace projection step is $\mathcal{O}(nr^4 + r^3 n\log n)$.

- **Stopping Criteria:** As indicated in (4.26), after projecting to Hankel space with $\mathcal{O}(r^2 n + rn\log n)$ flops via $\mathcal{H}^*$, we can compute $\|\partial F\|_F$ in $\mathcal{O}(n)$ flops. Thus, our stopping criteria can be efficiently verified with $\mathcal{O}(r^2 n + rn\log n)$ flops.

In summary, the computational complexity of our LPPG algorithm is $\mathcal{O}(r^4 n + r^3 n\log n)$ per iteration. The LPPG algorithm presents a highly efficient solution for larger scale SSS recovery, characterized by a manageable computational load even as the problem size increases. By ingeniously integrating the subspace projection and modified proximal gradient steps, it achieves a balance between computational demand and the ability to handle complex SSS recovery tasks. This efficiency, combined with its adaptability to various scenarios, including high-dimensional data, positions the LPPG algorithm as a significant advancement in the field of SCS.

Remark 11. *High-dimensional Extension The LPPG algorithm, informed by the Hankel-based methodologies in [2, 56–58], shows potential for adaptation to higher-dimensional SSS recovery. By leveraging the inherent Hankel structure in multidimensional SSS, our algorithm can be efficiently implemented for these complex scenarios. Key operations like Hankel matrix-vector multiplications and the use of the $\mathcal{H}^*$ operator are effectively applicable in these extended contexts, ensuring the adaptability and robustness of the LPPG algorithm for high-dimensional SSS recovery challenges.*

4.5 Simulations

This section presents numerical simulations to demonstrate the efficacy of the LPPG method, particularly focusing on its recovery capability under various conditions. The simulations are organized as follows:

1. **Phase Transition Analysis:** Section 4.5.1 evaluates the LPPG method's recovery ability using a phase transition framework, providing insights into its performance across different signal sparsity levels.

2. **Convergence Rate in Noise-Free Scenarios:** In Section 4.5.2, we examine the rapid convergence rate of LPPG in scenarios without noise, highlighting the benefits of an optimized step-size.

3. **Robustness to Heavy Noise:** Section 4.5.3 tests LPPG's performance in the presence of significant additive noise. This section also discusses the adaptability of the hyperparameter β and its impact on reconstruction quality.

4. **Mitigating Weighted Norm Bias:** Section 4.5.4 demonstrates how our proposed formulation effectively addresses the biased weighted norm issue, a common challenge in Hankel matrix-based methods.

5. **Sensitivity to Unknown Model Order:** In Section 4.5.5, the robustness of LPPG to variations in the model order is explored, underlining its effectiveness even when the exact number of signal components is not precisely known.

These simulations aim to comprehensively assess the LPPG method's performance under diverse conditions, verifying its practical applicability in signal recovery tasks.

In all tests, the ground-truth SSS and their partial observations are generated as follows: The frequency f_k of SSS is uniformly distributed on $[0, 1)$, and the phases of complex coefficients d_k are uniformly sampled on $[0, 2\pi)$. The amplitudes are set to $1 + 10^{0.5c_k}$, where c_k is uniformly distributed on $[0, 1]$; cf. [56]. For a given sample size $m = S_p n$, Ω is assumed to be uniformly

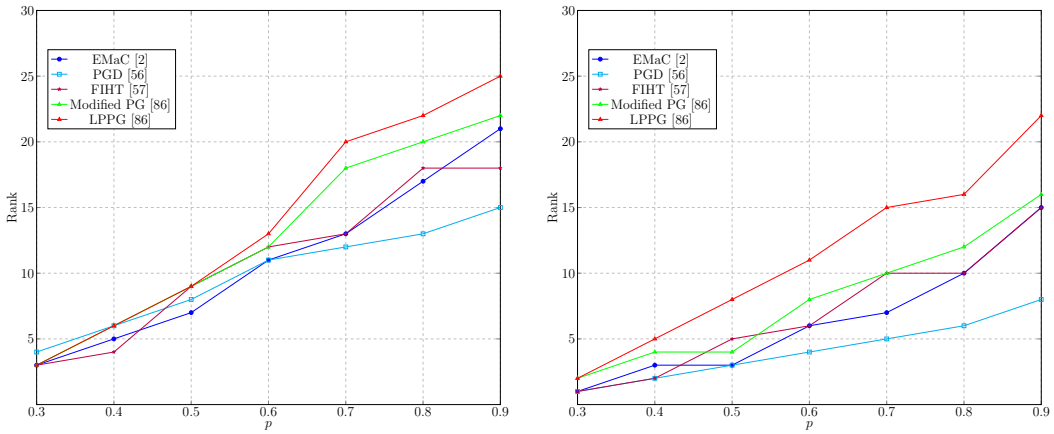
sampled. We assess computational efficiency (number of iterations) and normalized mean squared error (NMSE, defined as $\|\boldsymbol{x}^\dagger - \boldsymbol{x}\|_2/\|\boldsymbol{x}\|_2$, where $\boldsymbol{x}^\dagger$ denotes the estimate returned) to highlight the advantages of our LPPG method. We consider level-1, level-2, and level-3 cases in the data settings, with two different settings: (a) without damping factor, and (b) damping factors generated such that $1/\tau_{1,k}$ is uniformly sampled from $[8, 16]$ for $1 \leq k \leq r$ (similarly, $1/\tau_{2,k}$ from $[16, 32]$ and $1/\tau_{3,k}$ from $[64, 128]$) [56].

Our LPPG method is compared with benchmark algorithms, including EMaC [2], FIHT [57], PGD [56], standard PG [6], and the modified PG algorithm (MPG), which is LPPG without subspace projection. For large-size cases, we apply EMaC [2] using the alternating projection-based implementation as suggested in [2]. The value of α is set to $1e-20$ by default.

4.5.1 Empirical Phase Transition

The effectiveness of the recovery algorithms is quantified using the NMSE. A signal is considered successfully recovered if NMSE is less than 10^{-3}. Figure 4.1 presents the empirical phase transition, averaged over 50 trials. The phase transition curves, representing a 50% success rate, are plotted as a function of the sampling ratio S_p and rank r, for both undamped and damped cases. For computational feasibility, we set $n = 63$. The stopping criteria for the algorithms are limited to 1000 iterations (denoted as K) and a numerical-error threshold ϵ of 10^{-6}. It is noteworthy that the standard PG algorithm is excluded from these curves due to its non-convergence within the 1000 iteration limit.

Our simulations reveal that the damped case is inherently more challenging, requiring a higher sampling ratio for effective recovery compared to the undamped scenario. Notably, both the LPPG and MPG methods demonstrate superior performance over other algorithms in these contexts. The efficacy of the LPPG method is particularly pronounced in damped cases, highlighting the significance of its subspace projection step. This step is instrumental in ensuring successful recovery, especially when the model order is moderate relative to the signal size. The LPPG algorithm's ability to further reduce the objective function at each iteration is a key



(a) Without damping factor (b) With damping factor

Figure 4.1: The 50% phase transition curves: $n_1 = 63, (p_1, q_1) = (32, 32)$.

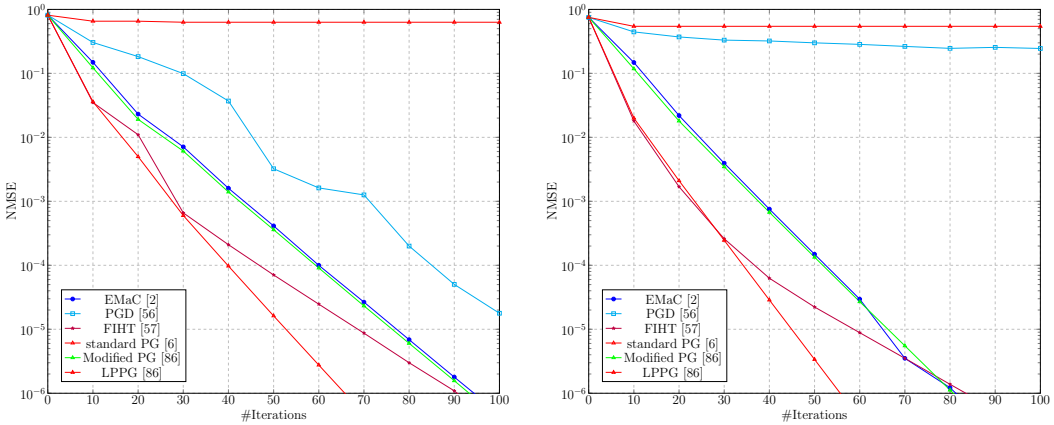
factor in its high performance. Crucially, the LPPG method shows an enhanced capacity for successful recovery with lower sampling ratios for a given rank r. This attribute is particularly beneficial in practical scenarios where efficient sensing is a priority. These outcomes strongly attest to the LPPG method's superiority in handling both damped and undamped spectrally sparse signals.

Remark 12. *While the LPPG method demonstrates superior recovery performance, we aim to maintain fairness in our comparative analysis. To this end, we consciously avoid scenarios exclusively favorable to LPPG in subsequent tests. Instead, we select conditions in Sections 4.5.2 and 4.5.3 that are amenable to all algorithms under consideration. This approach ensures a balanced comparison, reflecting each method's capabilities under a common set of challenges.*

4.5.2 Fast Convergence Rate for Noiseless Case

In this experiment, we evaluate the convergence rate of the LPPG algorithm in an ideal, noise-free scenario. The rate of convergence is closely linked to the choice of the parameter β. The theoretical bound on the convergence rate is given by the following inequality:

$$\min_{i=0,\cdots,K} \|\partial F(\boldsymbol{X}_{i+1}^e)\|_F^2 \leq \frac{(\beta - L_{\nabla f})\beta^4}{(K+1)(\alpha+\beta)^2}(F(\boldsymbol{X}_0^e) - F^*),$$



(a) Without damping factor and $S_p = 0.3$ (b) With damping factor and $S_p = 0.4$

Figure 4.2: Convergence rate comparison: $(n_1, n_2) = (31, 31), (p_1 p_2, q_1 q_2) = (256, 256), r = 15$. Left: no damping in the test signals and $S_p = 0.3$. Right: signals are generated with damping and $S_p = 0.4$.

suggesting that a smaller value of β results in a more rapid convergence. Following the recommendation in [88], in cases where the observed data is presumed accurate, setting the structural constraint parameter β near zero is advisable. Consequently, for the modified proximal gradient-based algorithms (MPG and LPPG), we set the step size $\gamma = 1/\beta = 1e6$. This choice ensures convergence for any β value, as $L_{\nabla f} < \beta$. In contrast, gradient-based methods such as FIHT and PGD require empirical adjustment of their step-size relative to the sampling ratio, as their step-size is not directly linked to the noise level. This necessitates a laborious tuning process. The alternating projection-based EMaC algorithm does not permit increasing the step size.

We compare the convergence rates of different algorithms in both undamped and damped settings in Figures 4.2a and 4.2b, respectively, by plotting NMSE as a function of the number of iterations. We test a two-dimensional SSS $\boldsymbol{x}$ of size 31×31, constructing a level-2 block Hankel matrix $\mathcal{H}\boldsymbol{x} \in \mathbb{C}^{256\times256}$. For the undamped setting, the sampling ratio is set to 0.3, and for the damped case, it is 0.4, with the rank consistently at 15. These parameters ensure that all algorithms are operable while allowing differentiation in their performance. The experiment is repeated over 50 trials with the stopping criteria of $K = 1000$ iterations and $\epsilon = 10^{-6}$.

Figures 4.2a and 4.2b display the convergence rates for the undamped and damped scenarios,

respectively.

- In both scenarios, the LPPG algorithm achieves the fastest convergence, clearly outperforming the other methods.
- The MPG algorithm also shows a quicker convergence compared to PGD and EMaC, underscoring the efficacy of the modified PG step.
- The standard PG algorithm lags in terms of convergence speed due to its smaller step size, dictated by $L_{\nabla f}$, highlighting the challenges in applying standard PG to SCS tasks with equally weighted norms.
- Although FIHT demonstrates competitive convergence speed, its final NMSE values exhibit instability due to the lack of guaranteed reduction in the objective function in each iteration.

These results confirm the significant impact of both the subspace projection and the modified PG step in ensuring faster convergence rates in SCS.

4.5.3 Robustness to Additive Noise

In practical applications, noise-free measurements are often unattainable. Therefore, the robustness of SSS recovery algorithms to additive noise is crucial. We introduce noise in the measurements with the following noise vector:

$$\boldsymbol{e} = \eta \cdot \|\mathcal{P}_\Omega \boldsymbol{x}\| \cdot \frac{\boldsymbol{w}}{\|\boldsymbol{w}\|},$$

where $\boldsymbol{w}$ is a standard complex Gaussian random vector, and η denotes the noise level. We focus on a challenging scenario with $\eta = 1$, corresponding to a Signal-to-Noise Ratio (SNR) of 0 dB.

We consider a three-dimensional SSS $\boldsymbol{x}$ of size $15 \times 15 \times 15$, with its Hankel matrix $\mathcal{H}\boldsymbol{x} \in \mathbb{C}^{512\times 512}$ forming a level-3 block. The sampling ratio is set to 1.0 in both undamped and damped

settings, emphasizing the denoising capability of the algorithms. The rank is fixed at 10. Results are averaged over 50 trials, providing a consistent and comparative evaluation of the algorithms' performance in the presence of significant noise. This experiment aims to test the algorithms' effectiveness in a noise-heavy environment, particularly focusing on the adaptability and performance of the LPPG approach. Given the challenging nature of a high noise level, this setup serves to demonstrate the utility of the Hankel enforcement parameter β in such conditions.

All algorithms tested in this experiment terminate under one of two conditions: either when the relative change in the Hankel matrix, represented as $\|\boldsymbol{X}_{k+1}^e - \boldsymbol{X}_k^e\|_F / \|\boldsymbol{X}_k^e\|_F$, falls below 10^{-3}, or when the maximum iteration count of 1000 is reached. This stopping criterion is deliberately set to a higher threshold due to the high noise level and the limited improvement in NMSE with an increasing number of iterations. The results of this test are summarized in Table 4.1.

The EMaC algorithm serves as a baseline for this experiment. It is a constrained optimization method implemented via alternating projections, and notably, it does not allow for variation in the Hankel enforcement parameter β. In contrast, to enhance reconstruction accuracy under noisy conditions, the LPPG method and other algorithms allow for a larger β value compared to noise-free scenarios. For this experiment, we select β values from the set $\{1 * \beta^*, 10\beta^*, 100\beta^*\}$[4], with the understanding that a larger β implies a smaller step size for the standard PG algorithm, potentially leading to its early termination. This setup not only tests the algorithms' robustness to noise but also demonstrates the impact of the parameter β on their performance.

The analysis of the results presented in Table 4.1 yields several crucial insights into the algorithms' performance under heavy additive noise:

1. *Enhanced Accuracy with Increased β:* A prominent trend observed is that larger values of the Hankel enforcement parameter (β) consistently lead to better recovery accuracy. This observation holds true across various algorithms, emphasizing the importance of adjusting β appropriately in noisy conditions.

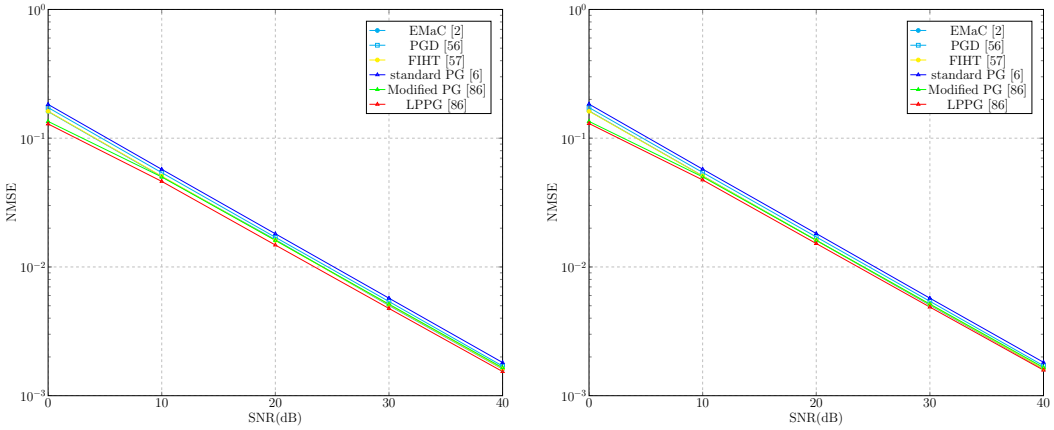
[4]Here the β^* is the default value for each algorithm. Moreover $\beta^* = S_p$ for Enhanced matrix form (PGD and FIHT), $\beta^* = S_p * n_1 n_2 n_3 / (p_1 p_2 p_3 * q_1 q_2 q_3)$ for vector form (PG-based algorithms) as a fair comparison.

Table 4.1: NMSE, number of iterations and time comparisons under heavy noise SNR = 0dB, $(n_1, n_2, n_3) = (15, 15, 15)$, $(p_1p_2p_3, q_1q_2q_3) = (512, 512)$, $r = 10$, $S_p = 1.0$.

β	$\beta = 1 * \beta^*$		$\beta = 10\beta^*$		$\beta = 100\beta^*$	
Criterion	NMSE	Iter	NMSE	Iter	NMSE	Iter
	undamped case					
EMaC [2]	0.161	7	0.161	12	0.161	12
PGD [56]	0.173	10	0.210	15	0.211	16
FIHT [57]	0.173	10	0.160	9	0.160	12
standard PG [6]	0.650	1	0.306	1	0.160	1
Modified PG [86]	0.646	7	0.277	18	0.136	37
LPPG [86]	0.646	6	0.277	12	0.128	31
	damped case					
EMaC [2]	0.163	7	0.163	12	0.163	12
PGD [56]	0.173	10	0.159	28	0.162	43
FIHT [57]	0.173	14	0.160	15	0.163	12
standard PG [6]	0.651	1	0.308	1	0.163	1
Modified PG [86]	0.646	7	0.278	23	0.134	43
LPPG [86]	0.646	7	0.278	20	0.129	36

2. *Superior Performance of MPG and LPPG with High β:* Notably, when β is set to 100, both the MPG and the LPPG methods exhibit superior recovery capabilities compared to other algorithms. In particular, LPPG achieves the highest accuracy levels. Interestingly, even the standard PG algorithm, with just one iteration, shows competitive performance under these conditions. On the other hand, algorithms like FIHT and PGD outperform EMaC only when β is increased, underscoring the pivotal role of the Hankel constraint in their performance.

3. *Effectiveness of Chosen Stopping Criteria:* The stopping criteria, particularly the term $\beta^2\mathcal{H}\mathcal{L}\mathcal{H}^*(\boldsymbol{X}^e_{k,\frac{1}{2}} - \boldsymbol{X}^e_{k+1}) \in \partial F(\boldsymbol{X}^e_{k+1})$, as discussed in Section 4.3.4, prove to be an effective measure for terminating the iterations of PG-based algorithms. This factor is especially crucial for methods like PGD and FIHT, which might otherwise demonstrate oscillatory behavior around the optimal solution.



(a) Without damping factor

(b) With damping factor

Figure 4.3: Performance for additive noise: $(n_1, n_2, n_3) = (15, 15, 15), (p_1p_2p_3, q_1q_2q_3) = (512, 512), r = 10, S_p = 1.0$. Left: no damping in the test signals. Right: signals are generated with damping.

Moreover, we show in Fig. 4.3 to demonstrate the outstanding performance to different levels of noise. The Hankel enforcement parameter β is optimized for each algorithm and noise level to pursue the best reconstruction.

These insights collectively underscore the robustness and efficiency of the LPPG method in the recovery of SSSs amidst heavy additive noise. This robustness, combined with the algorithm's computational efficiency, highlights its potential applicability in real-world scenarios where noise is a prevalent challenge.

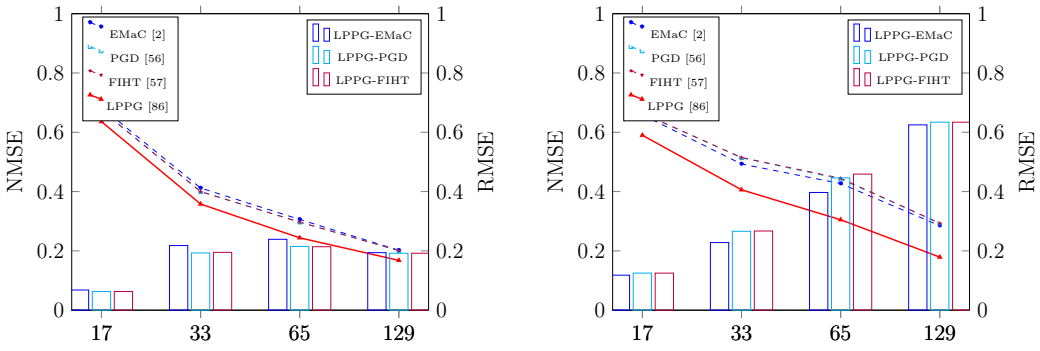
4.5.4 The Benefit of Vector Formulation

Our research makes a significant contribution by adapting PG algorithms for vector formulations, particularly through the introduction of the modified PG approach. This section highlights the benefits of our vector form formulations, which align closely with the original problem and improve reconstruction accuracy as the signal size increases.

Previous studies, such as those by Gillard et al. [119] and Wang et al. [58], have noted that traditional algorithms like Cadzow's may result in biased reconstructions due to the presence of repetitive elements in the enhanced matrix. To illustrate this issue, we analyze the Normalized

NMSE for several algorithms—FIHT, PGD, EMaC, and our LPPG method (with MPG and standard PG sharing the same data fidelity item as LPPG). We focus on one-dimensional Spectrally Sparse Signal (1D SSS) recovery for varying signal sizes $n \in \{17, 33, 65, 129\}$, examining how the impact of repetitive elements escalates with increasing signal size. The NMSE result, averaged over 50 trials, consider fully observed signals with added noise and a rank of 2. The stopping criteria are set at a maximum of 1000 iterations ($K = 1000$) and a numerical-error threshold of 10^{-6} ($\epsilon = 10^{-6}$). The Hankel enforcement parameter β is optimized for each algorithm and noise level to pursue the best reconstruction.

Furthermore, we present a comparative analysis using a bar chart to depict the relative mean squared error (RMSE) in relation to LPPG. The RMSE is defined as $\|\text{NMSE}_{\text{LPPG}} - \text{NMSE}\|_2/\|\text{NMSE}_{\text{LPPG}}\|_2$, where $\text{NMSE}_{\text{LPPG}}$ represents the NMSE achieved by the LPPG algorithm. This comparison aims to provide a clear metric for evaluating the recovery capability of different algorithms across various signal sizes.



(a) Without damping factor (b) With damping factor

Figure 4.4: Convergence rate comparison: $n_1 = 17, 33, 65, 129$, $p_1 = q_1 = 9, 17, 33, 65$, $r = 2$, $S_p = 1.0$,SNR= 0dB.

The results depicted in Figure 4.4 lead us to two essential observations regarding the performance of the tested algorithms in 1D SSS recovery, particularly under the influence of signal size and algorithmic approach.

1. **Improved Reconstruction with Increasing Signal Size:** All algorithms under consideration exhibit enhanced reconstruction accuracy as the signal size grows. This improvement is likely a consequence of the signal's increased sparsity in the frequency

domain, which becomes more pronounced with larger signal sizes. This trend is observed consistently across different algorithms, indicating a general characteristic of SSS recovery approaches.

2. **Consistent Superiority of the LPPG Algorithm:** The LPPG algorithm stands out for its consistently superior performance across all tested signal sizes. Notably, the RMSE in comparison to LPPG increases for other algorithms as the signal size grows. This trend underscores the LPPG algorithm's robustness against the impact of repetitive elements in the enhanced matrix, which becomes more significant in larger signal sizes. Furthermore, the advantage of the LPPG algorithm is particularly remarkable in cases involving damped signals, emphasizing its effectiveness in handling more complex SSS recovery scenarios.

These observations affirm the efficacy of the LPPG algorithm in SSS recovery tasks, highlighting its capability to overcome challenges posed by larger signal sizes and the presence of repetitive elements in the enhanced matrix.

4.5.5 Sensitivity to Model Order

In practical scenarios, the precise model order of a SSS might not be accurately known. This section discusses the robustness of the LPPG method, alongside other algorithms, under scenarios of both under- and over-estimation of the model order. We conducted experiments on three-dimensional signals with an actual model order of $r = 5$ and a sampling rate of 0.5, covering both undamped and damped signals. Our analysis focuses on the NMSE and the number of iterations (Iter) required for convergence. The stopping criteria are set at a maximum of 1000 iterations ($K = 1000$) and a numerical-error threshold of 10^{-6} ($\epsilon = 10^{-6}$). The Hankel enforcement parameter β is optimized for each algorithm and noise level to pursue the best reconstruction.

1. **Optimal NMSE at True Model Order:** All algorithms exhibit the lowest NMSE when the input model order r matches the true model order of 5.

2. **Large NMSE Under-Estimation or Over-Estimation:** Under-estimation or Over-estimation of r results in larger NMSE due to systematic truncation errors.

3. **Desirable NMSE with Over-Estimation:** Increasing the model order beyond the true value maintains desirable NMSE, albeit at the cost of significantly more iterations, particularly evident in algorithms other than LPPG.

4. **Efficient Over-Estimation with LPPG:** LPPG shows an ability to achieve good NMSE even with over-estimated r, without incurring excessive additional iterations compared to the scenario with the true model order.

Results are summarized in Tables 4.2 and 4.3 for undamped and damped signals, respectively, averaged over 50 random instances. These findings indicate two feasible strategies for LPPG when the exact model order is unknown:

- *Rank Increasing Heuristic:* Begin with a low rank, increase it incrementally, executing the LPPG algorithm until convergence at each step, and monitor the NMSE on the observed entries. The process continues until no significant improvement in NMSE is observed with an increase in rank [56, 57].

- *Slight Over-Estimation Approach:* If reconstruction accuracy under heavy noise is not the primary concern, a straightforward approach is to slightly over-estimate the model order and run the LPPG algorithm.

These strategies provide a practical guide for applying the LPPG method effectively, even in the absence of precise information on the model order of the underlying SSS.

4.6 Conclusions

In this chapter, we presented the low-rank projected proximal gradient method, an innovative approach for SSS recovery. Central to this approach is a non-convex optimization framework

Table 4.2: Mean values of NMSE and Iter over 50 undamped random instances: $\boldsymbol{X} \in \mathbb{C}^{11\times11\times11}$, $\boldsymbol{X}^e \in \mathbb{C}^{216\times216}$ and $S_p = 0.5$.

Test rank		2	3	4	5	6	7	8
	SNR = ∞							
FIHT [57]	NMSE	0.66	0.47	0.29	2.6e-7	1.0e-2	1.4e-2	1.9e-2
	Iter	12	13	13	14	1000	1000	1000
PGD [56]	NMSE	0.66	0.47	0.29	6.6e-6	2.3e-5	8.0e-4	1.0e-3
	Iter	278	96	154	53	872	1000	1000
Modified PG [86]	NMSE	0.50	0.34	0.22	1.0e-6	4.5e-4	1.3e-3	1.5e-3
	Iter	25	26	28	29	53	53	52
LPPG [86]	NMSE	0.50	0.34	0.22	1.0e-6	1.0e-6	1.0e-6	1.0e-6
	Iter	17	18	19	17	18	18	18
	SNR = 20							
FIHT [57]	NMSE	0.67	0.49	0.31	2.2e-2	2.4e-2	2.9e-2	3.3e-2
	Iter	102	102	125	124	281	292	732
PGD [56]	NMSE	0.67	0.49	0.37	2.2e-2	2.2e-2	2.3e-2	2.8e-2
	Iter	215	309	429	351	1000	1000	1000
Modified PG [86]	NMSE	0.63	0.46	0.29	1.9e-2	2.1e-2	2.5e-2	2.7e-2
	Iter	366	354	392	386	566	443	1000
LPPG [86]	NMSE	0.63	0.46	0.29	1.9e-2	2.1e-2	2.5e-2	2.8e-2
	Iter	129	136	144	176	209	227	301
	SNR = 0							
FIHT [57]	NMSE	0.67	0.51	0.36	0.22	0.26	0.30	0.33
	Iter	108	117	123	119	724	1000	1000
PGD [56]	NMSE	0.67	0.51	0.36	0.22	0.40	0.30	0.33
	Iter	289	312	444	356	1000	1000	1000
Modified PG [86]	NMSE	0.65	0.49	0.34	0.18	0.21	0.25	0.27
	Iter	377	378	382	380	1000	1000	1000
LPPG [86]	NMSE	0.65	0.49	0.34	0.16	0.21	0.29	0.29
	Iter	172	175	189	192	556	566	581

that incorporates an enhanced low-rank Hankel matrix while evaluating the residual error in the vector form of the original signal. This vector formulation, shared with two other algorithms proposed in this thesis, demonstrates broad applicability and benefits, as shown in Section 4.5.4.

The core innovation of the LPPG method lies in its iterative process, which comprises two

Table 4.3: Mean values of NMSE and Iter over 50 damped random instances: $\boldsymbol{x} \in \mathbb{C}^{11\times11\times11}$, $\boldsymbol{X}^e \in \mathbb{C}^{216\times216}$ and $S_p = 0.5$.

Test rank		2	3	4	5	6	7	8
	SNR = ∞							
FIHT [57]	NMSE	0.56	0.38	0.21	1.1e-7	1.1e-2	2.4e-2	4.6e-2
	Iter	10	11	13	16	1000	1000	1000
PGD [56]	NMSE	0.56	0.38	0.21	8.6e-6	2.0e-4	2.3e-3	1.9e-3
	Iter	415	123	205	168	1000	1000	1000
Modified PG [86]	NMSE	0.41	0.28	0.15	8.5e-7	3.9e-3	6.4e-3	8.0e-3
	Iter	20	19	25	26	245	465	521
LPPG [86]	NMSE	0.41	0.28	0.15	1.0e-6	1.0e-6	1.0e-6	1.0e-6
	Iter	13	16	17	18	18	18	19
	SNR = 20							
FIHT [57]	NMSE	0.59	0.38	0.22	2.3e-2	2.7e-2	2.9e-2	3.5e-2
	Iter	110	113	121	123	444	453	598
PGD [56]	NMSE	0.59	0.38	0.22	2.3e-2	2.3e-2	2.3e-2	2.3e-2
	Iter	356	479	732	1000	1000	1000	1000
Modified PG [86]	NMSE	0.56	0.36	0.20	1.9e-2	2.2e-2	3.3e-2	3.0e-2
	Iter	376	389	389	395	427	1000	516
LPPG [86]	NMSE	0.56	0.36	0.20	1.9e-2	2.2e-2	2.3e-2	2.4e-2
	Iter	115	148	164	169	179	193	194
	SNR = 0							
FIHT [57]	NMSE	0.60	0.41	0.29	0.23	0.26	0.30	0.33
	Iter	119	129	134	142	867	672	685
PGD [56]	NMSE	0.60	0.41	0.36	0.23	0.24	0.26	0.28
	Iter	394	583	965	853	1000	10000	1000
Modified PG [86]	NMSE	0.58	0.39	0.27	0.19	0.22	0.24	0.27
	Iter	384	386	398	415	1000	1000	1000
LPPG [86]	NMSE	0.58	0.39	0.27	0.19	0.22	0.24	0.27
	Iter	147	178	212	212	643	645	656

distinct steps in each iteration. The first step involves optimization with respect to all variables, excluding the column and row sub-spaces of the Hankel matrix. The second step simultaneously updates these sub-spaces along with all other variables. Each step is meticulously tailored to leverage the inherent structures of the problem, optimizing the recovery process.

A notable feature of the LPPG method is its theoretical guarantee of convergence. This characteristic not only solidifies its theoretical foundation but also significantly enhances its practical applicability. Extensive numerical simulations have underscored the LPPG method's superiority, showcasing a faster convergence rate and higher reconstruction accuracy compared to existing benchmark algorithms. This performance is especially pronounced in scenarios characterized by larger scale data and substantial noise levels.

Overall, the LPPG method emerges as a robust and efficient solution for SSS recovery, adept at tackling the complexities and challenges prevalent in practical settings.

This chapter has successfully extended the PG method for vector-form CS, culminating in the development of the LPPG method. Despite its notable successes, areas for further improvement remain.

1. **Computational Complexity:** The primary computational load in LPPG arises from the low-rank subspace projection. This aspect becomes particularly challenging for larger scale signals with high model orders, limiting the method's practicality in such scenarios.
2. **Balance Between Convergence and Accuracy:** While the low-rank projection effectively reduces iteration count, it remains significant, especially under larger Hankel enforcement parameters. This necessitates a careful trade-off between convergence rate and reconstruction accuracy, dictated by the choice of the Hankel enforcement parameter. This is emperically solved by early termination of the algorithm since the NMSE does not improve significantly after a certain number of iterations.

Looking forward, the integration of accelerated proximal gradient techniques into the LPPG framework presents an exciting avenue for research. Such advancements could yield more efficient and potent solutions for SSS recovery. The next two chapters of this thesis will explore these possibilities, aiming to further refine and enhance the capabilities of the PG method in handling complex signal processing challenges. Overall, the journey of enhancing the LPPG method continues, with the goal of achieving optimal balance in computational efficiency, convergence rate, and reconstruction accuracy for larger scale SSS recovery tasks.

4.7 Proofs

In this section, we provide mathematical proofs for the theoretical results stated in Proposition 1, Proposition 2, Proposition 3, and Theorem 1.

We begin by proving the following lemma, which offers a useful formula for computing the gradient of a function minimized with respect to its arguments.

Lemma 2. *Let $h : \mathbb{C}^{p\times q} \times \mathbb{C}^n \to \mathbb{R}$ be a continuously differentiable function, and let $f : \mathbb{C}^{p\times q} \to \mathbb{R}$ be defined as $f(\boldsymbol{X}^e) = \min_{\boldsymbol{x}\in\mathbb{C}^n} h(\boldsymbol{X}^e, \boldsymbol{x})$. Assume that for each $\boldsymbol{X}^e \in \mathbb{C}^{p\times q}$, there exists a minimizer $\boldsymbol{x}^\star(\boldsymbol{X}^e)$.*

Then, for any $\boldsymbol{X}^e \in \mathbb{C}^{p\times q}$, the gradient of f with respect to $\boldsymbol{X}^e$ exists and is given by:

$$\nabla f(\boldsymbol{X}^e) = \nabla_{\boldsymbol{X}^e} h(\boldsymbol{X}^e, \boldsymbol{x}^\star),$$

Proof. To prove this lemma, we utilize the chain rule of calculus. Let $\boldsymbol{x}^\star$ be the minimizer of $h(\boldsymbol{X}^e, \boldsymbol{x})$, such that $f(\boldsymbol{X}^e) = h(\boldsymbol{X}^e, \boldsymbol{x}^\star)$. We then have:

$$\begin{aligned}
\nabla f(\boldsymbol{X}^e) &= \nabla h(\boldsymbol{X}^e, \boldsymbol{x}^\star) \\
&= \nabla_{\boldsymbol{X}^e} h(\boldsymbol{X}^e, \boldsymbol{x}^\star) + \nabla_{\boldsymbol{x}} h(\boldsymbol{X}^e, \boldsymbol{x}^\star) \nabla_{\boldsymbol{X}^e} \boldsymbol{x}^\star \\
&= \nabla_{\boldsymbol{X}^e} h(\boldsymbol{X}^e, \boldsymbol{x}^\star),
\end{aligned}$$

where the second equality is derived from the chain rule, and the third equality follows from the optimality condition $\nabla_{\boldsymbol{x}} h(\boldsymbol{X}^e, \boldsymbol{x}^\star) = \boldsymbol{0}$.

Hence, we have proven that $\nabla f(\boldsymbol{X}^e) = \nabla_{\boldsymbol{X}^e} h(\boldsymbol{X}^e, \boldsymbol{x}^\star)$. □

4.7.1 Proof of Prop. 1

We adapted a modification to the non-convex non-smooth unconstrained optimization problem by transforming it into a single variable problem $F(\boldsymbol{X}^e) = f(\boldsymbol{X}^e) + g(\boldsymbol{X}^e)$. A key advantage

of this approach is achieving a better upper bound for $L_{\nabla f}$ as β, which is independent of the signal size. The proof is as follows:

Proof. Firstly, consider the gradient with respect to $\boldsymbol{x}$:

$$\begin{aligned}&\nabla_{\boldsymbol{x}}\big(\frac{1}{2}\|\boldsymbol{s}-\mathcal{P}_\Omega\boldsymbol{x}\|^2+\frac{\beta}{2}\|\boldsymbol{X}^e-\mathcal{H}\boldsymbol{x}\|_F^2+\frac{\alpha}{2}\|\boldsymbol{x}\|^2\big)\\&=\underbrace{(\alpha\mathcal{I}+\mathcal{P}_\Omega^*\mathcal{P}_\Omega+\beta\mathcal{H}^*\mathcal{H})^{-1}}_{\mathcal{L}}\boldsymbol{x}-\mathcal{P}_\Omega^*\boldsymbol{s}-\beta\mathcal{H}^*\boldsymbol{X}^e.\end{aligned}\tag{4.30}$$

Setting this gradient to zero yields the closed-form solution for $\boldsymbol{x}^\star$:

$$\boldsymbol{x}^\star=\mathcal{L}(\mathcal{P}_\Omega^*\boldsymbol{s}+\beta\mathcal{H}^*\boldsymbol{X}^e).$$

Substituting $\boldsymbol{x}^\star$ into $h(\boldsymbol{X}^e,\boldsymbol{x})$, we get:

$$f(\boldsymbol{X}^e)=h(\boldsymbol{X}^e,\boldsymbol{x}^\star)=\frac{1}{2}\|\boldsymbol{s}-\mathcal{P}_\Omega\boldsymbol{x}^\star\|^2+\frac{\beta}{2}\|\boldsymbol{X}^e-\mathcal{H}\boldsymbol{x}^\star\|_F^2+\frac{\alpha}{2}\|\boldsymbol{x}\|^2.$$

Applying Lemma 2, we have:

$$\begin{aligned}\nabla f(\boldsymbol{X}^e)&=\nabla_{\boldsymbol{X}^e}h(\boldsymbol{X}^e,\boldsymbol{x}^\star)=\beta(\boldsymbol{X}^e-\mathcal{H}\boldsymbol{x}^\star)\\&=\beta\boldsymbol{X}^e-\beta\mathcal{H}\mathcal{L}(\mathcal{P}_\Omega^*\boldsymbol{s}+\beta\mathcal{H}^*\boldsymbol{X}^e)\end{aligned}\tag{4.31}$$

The Hessian matrix of $f(\boldsymbol{X}^e)$ is then $\beta\mathcal{I}-\beta^2\mathcal{H}\mathcal{L}\mathcal{H}^*$. Since:

$$0\le\|\beta^2\mathcal{H}\mathcal{L}\mathcal{H}^*\|=\beta\|\beta\mathcal{H}(\alpha\mathcal{I}+\mathcal{P}_\Omega\mathcal{P}_\Omega^*+\beta\mathcal{H}^*\mathcal{H})^{-1}\mathcal{H}^*\|<\beta,\tag{4.32}$$

we conclude that $L_{\nabla f}<\beta$.[5] □

[5]The operator norm is used in (4.32). It is the maximum eigenvalue of the corresponding matrix representation.

4.7.2 Proof of Prop. 2

In this proof, we aim to efficiently solve the low-rank subspace projection step by transforming $h(\boldsymbol{\Sigma}, \boldsymbol{x})$ into a single variable function $f(\boldsymbol{\Sigma})$. We will detail the procedure to find the optimal value for the low-rank subspace projection.

Proof. Consider the optimization problem:

$$\begin{aligned} L(\boldsymbol{\Sigma}) &= \min_{\boldsymbol{x}} \; h(\boldsymbol{\Sigma}, \boldsymbol{x}) \\ &= \min_{\boldsymbol{x}} \; \frac{1}{2}\|\boldsymbol{s} - \mathcal{P}_\Omega \boldsymbol{x}\|^2 + \frac{\beta}{2}\|\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathsf{H}} - \mathcal{H}\boldsymbol{x}\|_F^2 + \frac{\alpha}{2}\|\boldsymbol{x}\|^2. \end{aligned}$$

The optimal $\boldsymbol{x}$, denoted as $\boldsymbol{x}^\dagger$, for any given $\boldsymbol{\Sigma}$, is found by solving the inner least squares problem $h(\boldsymbol{\Sigma}, \boldsymbol{x})$.

By differentiating with respect to $\boldsymbol{x}$, we obtain:

$$\frac{\partial}{\partial \boldsymbol{x}} h(\boldsymbol{\Sigma}, \boldsymbol{x}) = (\alpha\mathcal{I} + \mathcal{P}_\Omega^* \mathcal{P}_\Omega + \beta\mathcal{H}^*\mathcal{H})\boldsymbol{x} - \mathcal{P}_\Omega^* \boldsymbol{s} - \beta\mathcal{H}^* \mathcal{P}_{\boldsymbol{U},\boldsymbol{V}} \boldsymbol{\Sigma}. \tag{4.33}$$

Setting (4.33) to zero yields the closed-form solution for $\boldsymbol{x}^\dagger$:

$$\boldsymbol{x}^\dagger = \underbrace{(\alpha\mathcal{I} + \mathcal{P}_\Omega^* \mathcal{P}_\Omega + \beta\mathcal{H}^*\mathcal{H})^{-1}}_{\mathcal{L}} (\mathcal{P}_\Omega^* \boldsymbol{s} + \beta\mathcal{H}^* \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathsf{H}}).$$

Applying Lemma 2, the gradient of f with respect to $\boldsymbol{\Sigma}$ is:

$$\begin{aligned} \nabla f(\boldsymbol{\Sigma}) &= \nabla_{\boldsymbol{\Sigma}} h(\boldsymbol{\Sigma}, \boldsymbol{x}^\dagger) \\ &= \beta \mathcal{P}_{\boldsymbol{U},\boldsymbol{V}}^* \mathcal{P}_{\boldsymbol{U},\boldsymbol{V}} \boldsymbol{\Sigma} - \beta \mathcal{P}_{\boldsymbol{U},\boldsymbol{V}}^* \mathcal{H}\boldsymbol{x}^\dagger \\ &= \beta\boldsymbol{\Sigma} - \beta \mathcal{P}_{\boldsymbol{U},\boldsymbol{V}}^* \mathcal{H}\mathcal{L}(\mathcal{P}_\Omega^* \boldsymbol{s} + \beta\mathcal{H}^* \mathcal{P}_{\boldsymbol{U},\boldsymbol{V}} \boldsymbol{\Sigma}). \end{aligned} \tag{4.34}$$

The solution to the linear equation

$$[\mathcal{I} - \beta \mathcal{P}^*_{\boldsymbol{U},\boldsymbol{V}} \mathcal{H}\mathcal{L}\mathcal{H}^* \mathcal{P}_{\boldsymbol{U},\boldsymbol{V}}]\boldsymbol{\Sigma} = \mathcal{P}^*_{\boldsymbol{U},\boldsymbol{V}} \mathcal{H}\mathcal{L}\mathcal{P}^*_{\Omega}\boldsymbol{s}. \tag{4.35}$$

provides the Proposition 2 results:

$$\begin{aligned} \boldsymbol{x}^\star &= (\alpha\mathcal{I} + \mathcal{P}^*_{\Omega}\mathcal{P}_{\Omega} + \beta\mathcal{H}^*\mathcal{H})^{-1}(\mathcal{P}^*_{\Omega}\boldsymbol{s} + \beta\mathcal{H}^*\mathcal{P}_{\boldsymbol{U},\boldsymbol{V}}\boldsymbol{\Sigma}^\star), \\ \boldsymbol{\Sigma}^\star &= \left(\mathcal{I} - \beta\mathcal{P}^*_{\boldsymbol{U},\boldsymbol{V}}\mathcal{H}\mathcal{L}\mathcal{H}^*\mathcal{P}_{\boldsymbol{U},\boldsymbol{V}}\right)^{-1}\left(\mathcal{P}^*_{\boldsymbol{U},\boldsymbol{V}}\mathcal{H}\mathcal{L}\mathcal{P}^*_{\Omega}\boldsymbol{s}\right). \end{aligned} \tag{4.36}$$

□

4.7.3 Proof of Prop. 3

The proposition asserts that the low-rank subspace projection step guarantees sufficient decrease in the objective function. The formal statement is as follows:

$$F(\boldsymbol{X}^e_{k,\frac{1}{2}}) \leq F(\boldsymbol{X}^e_k) - \left(\frac{\alpha}{\alpha+\beta}\right)\|\boldsymbol{X}^e_{k,\frac{1}{2}} - \boldsymbol{X}^e_k\|^2_F. \tag{4.37}$$

Proof. Define the mapping $\mathcal{M}$ as follows:

$$\mathcal{M} = \left(\mathcal{I} - \beta\mathcal{P}^*_{\boldsymbol{U},\boldsymbol{V}}\mathcal{H}\mathcal{L}\mathcal{H}^*\mathcal{P}_{\boldsymbol{U},\boldsymbol{V}}\right). \tag{4.38}$$

Then, consider the following inequality chain:

$$\|\boldsymbol{X}^e_{k,\frac{1}{2}} - \boldsymbol{X}^e_k\|^2_F = \|\boldsymbol{\Sigma}_{k,\frac{1}{2}} - \boldsymbol{\Sigma}_k\|^2_F \tag{4.39}$$

$$\overset{(a)}{\leq} \left(\frac{\alpha+\beta}{\alpha}\right)\|\boldsymbol{\Sigma}_{k,\frac{1}{2}} - \boldsymbol{\Sigma}_k\|^2_{\mathcal{M}}$$

$$\overset{(b)}{\leq} \left(\frac{\alpha+\beta}{\alpha}\right)\left(F(\boldsymbol{X}^e_k) - F(\boldsymbol{X}^e_{k,\frac{1}{2}})\right). \tag{4.40}$$

Inequality (a) follows from the fact that $\frac{\alpha}{\alpha+\beta}$ is the smallest eigenvalue of the positive definite mapping $\mathcal{M}$. Inequality (b) arises from the optimal solution of the least squares problem in the

low-rank subspace projection step. □

4.7.4 Proof of Theorem 1

Our non-convex PG-based methods, featuring two monotonously decreasing steps, converge to critical points. This convergence is primarily attributed to the modified PG step, which demonstrates that $\|\partial F(\boldsymbol{X}^e_{k+1})\|_F \leq (1/\gamma + L_{\nabla f})\|\boldsymbol{X}^e_{k+1} - \boldsymbol{X}^e_{k,\frac{1}{2}}\|_F$, with the latter term approaching zero. We also provide a convergence speed analysis. The proof can be divided into three key steps:

1. **Sufficient Decrease of the PG Step:** In Algorithm 6, observe that:

$$\begin{aligned}\boldsymbol{X}^e_{k+1} = \arg\min_{\boldsymbol{X}^e} &\langle \nabla f(\boldsymbol{X}^e_{k,\frac{1}{2}}), \boldsymbol{X}^e - \boldsymbol{X}^e_{k,\frac{1}{2}}\rangle \\ &+ \frac{1}{2\gamma}\|\boldsymbol{X}^e - \boldsymbol{X}^e_{k,\frac{1}{2}}\|_F^2 + g(\boldsymbol{X}^e).\end{aligned} \tag{4.41}$$

Therefore, we have:

$$\begin{aligned}&\langle \nabla f(\boldsymbol{X}^e_{k,\frac{1}{2}}), \boldsymbol{X}^e_{k+1} - \boldsymbol{X}^e_{k,\frac{1}{2}}\rangle \\ &+ \frac{1}{2\gamma}\|\boldsymbol{X}^e_{k+1} - \boldsymbol{X}^e_{k,\frac{1}{2}}\|_F^2 + g(\boldsymbol{X}^e_{k+1}) \\ &\leq g(\boldsymbol{X}^e_{k,\frac{1}{2}}).\end{aligned} \tag{4.42}$$

From the Lipschitz continuous of ∇f we have

$$
\begin{aligned}
F(\boldsymbol{X}^e_{k+1}) &\leq g(\boldsymbol{X}^e_{k+1}) + f(\boldsymbol{X}^e_{k,\frac{1}{2}}) \qquad (4.43)\\
&+ \langle \nabla f(\boldsymbol{X}^e_{k,\frac{1}{2}}), \boldsymbol{X}^e_{k+1} - \boldsymbol{X}^e_{k,\frac{1}{2}} \rangle + \frac{L_{\nabla f}}{2} \|\boldsymbol{X}^e_{k+1} - \boldsymbol{X}^e_{k,\frac{1}{2}}\|_F^2 \\
&\leq g(\boldsymbol{X}^e_{k,\frac{1}{2}}) - \langle \nabla f(\boldsymbol{X}^e_{k,\frac{1}{2}}), \boldsymbol{X}^e_{k+1} - \boldsymbol{X}^e_{k,\frac{1}{2}} \rangle \\
&- \frac{1}{2\gamma} \|\boldsymbol{X}^e_{k+1} - \boldsymbol{X}^e_{k,\frac{1}{2}}\|_F^2 + f(\boldsymbol{X}^e_{k,\frac{1}{2}}) \\
&+ \langle \nabla f(\boldsymbol{X}^e_{k,\frac{1}{2}}), \boldsymbol{X}^e_{k+1} - \boldsymbol{X}^e_{k,\frac{1}{2}} \rangle - \frac{L_{\nabla f}}{2} \|\boldsymbol{X}^e_{k+1} - \boldsymbol{X}^e_{k,\frac{1}{2}}\|_F^2 \\
&= F(\boldsymbol{X}^e_{k,\frac{1}{2}}) - \frac{\frac{1}{\gamma} - L_{\nabla f}}{2} \|\boldsymbol{X}^e_{k+1} - \boldsymbol{X}^e_{k,\frac{1}{2}}\|_F^2. \qquad (4.44)
\end{aligned}
$$

2. H^* **is a critical point:** $0 \in \partial F(\boldsymbol{X}^{e*})$

Due to the decrease property of the subspace projection step, we have:

$$
F(\boldsymbol{X}^e_{k+1,\frac{1}{2}}) \leq F(\boldsymbol{X}^e_{k+1}) \leq F(\boldsymbol{X}^e_{k,\frac{1}{2}}). \qquad (4.45)
$$

Thus:

$$
F(\boldsymbol{X}^e_{k+1,\frac{1}{2}}) \leq F(\boldsymbol{X}^e_0), \quad F(\boldsymbol{X}^e_{k+1}) \leq F(\boldsymbol{X}^e_0) \qquad (4.46)
$$

for all k. It is evident that $\{\boldsymbol{X}^e_k\}$ and $\{\boldsymbol{X}^e_{k,\frac{1}{2}}\}$ are bounded. Therefore, $\{\boldsymbol{X}^e_k\}$ has accumulation points. As $F(\boldsymbol{X}^e_k)$ is decreasing, F has the same value at all the accumulation points. Let this value be F^*. Based on (4.44), we have:

$$
\begin{aligned}
\frac{\frac{1}{\gamma} - L_{\nabla f}}{2} \|\boldsymbol{X}^e_{k+1} - \boldsymbol{X}^e_{k,\frac{1}{2}}\|_F^2 &\leq F(\boldsymbol{X}^e_{k,\frac{1}{2}}) - F(\boldsymbol{X}^e_{k+1}) \\
&\leq F(\boldsymbol{X}^e_k) - F(\boldsymbol{X}^e_{k+1}). \qquad (4.47)
\end{aligned}
$$

Summing over $k = 0, \ldots, K$, we obtain:

$$
\begin{aligned}
&\frac{\frac{1}{\gamma} - L_{\nabla f}}{2} \sum_{k=0}^{K} \|\boldsymbol{X}^e_{k+1} - \boldsymbol{X}^e_{k,\frac{1}{2}}\|_F^2 \\
&\leq F(\boldsymbol{X}^e_0) - F(\boldsymbol{X}^e_k) \leq F(\boldsymbol{X}^e_0) - F^* \leq \infty. \qquad (4.48)
\end{aligned}
$$

As $L_{\nabla f} < \frac{1}{\gamma}$, we have:

$$\lim_{k\to\infty} \|\boldsymbol{X}^e_{k+1} - \boldsymbol{X}^e_{k,\frac{1}{2}}\|_F^2 \to 0. \tag{4.49}$$

From the optimality condition of (4.41), we have:

$$0 \in \nabla f(\boldsymbol{X}^e_{k,\frac{1}{2}}) + \frac{1}{\gamma}(\boldsymbol{X}^e_{k+1} - \boldsymbol{X}^e_{k,\frac{1}{2}}) + \partial g(\boldsymbol{X}^e_{k+1}). \tag{4.50}$$

Hence:

$$-\nabla f(\boldsymbol{X}^e_{k,\frac{1}{2}}) + \nabla f(\boldsymbol{X}^e_{k+1}) - \frac{1}{\gamma}(\boldsymbol{X}^e_{k+1} - \boldsymbol{X}^e_{k,\frac{1}{2}}) \in \partial F(\boldsymbol{X}^e_{k+1}). \tag{4.51}$$

Moreover, we get the key result:

$$\boxed{\begin{aligned} &\| - \nabla f(\boldsymbol{X}^e_{k,\frac{1}{2}}) + \nabla f(\boldsymbol{X}^e_{k+1}) - \frac{1}{\gamma}(\boldsymbol{X}^e_{k+1} - \boldsymbol{X}^e_{k,\frac{1}{2}})\| \\ &\leq (\frac{1}{\gamma} + L_{\nabla f})\|\boldsymbol{X}^e_{k+1} - \boldsymbol{X}^e_{k,\frac{1}{2}}\| \to 0 \quad \text{as} \quad k \to \infty. \end{aligned}} \tag{4.52}$$

Let $\boldsymbol{X}^{e*}$ be any accumulation point of $\{\boldsymbol{X}^e_{k,\frac{1}{2}}\}$, say $\{\boldsymbol{X}^e_{k_j,\frac{1}{2}}\} \to \boldsymbol{X}^{e*}$ as $j \to \infty$. From (4.49), we have $\{\boldsymbol{X}^e_{k_j+1}\} \to \boldsymbol{X}^{e*}$ as $j \to \infty$. Since f is continuously differentiable and g is lower semicontinuous, we have:

$$\lim_{j\to\infty} F(\boldsymbol{X}^e_{k_j+1}) = F(\boldsymbol{X}^{e*}). \tag{4.53}$$

Therefore, combining $\{\boldsymbol{X}^e_{k_j+1}\} \to \boldsymbol{X}^{e*}$, (4.22), (4.23), and (4.53), we conclude:

$$0 \in \partial F(\boldsymbol{X}^{e*}). \tag{4.54}$$

3. **Convergence speed** Given that (4.26) implies $\beta^2\mathcal{H}\mathcal{L}\mathcal{H}^*(\boldsymbol{X}^e_{k,\frac{1}{2}} - \boldsymbol{X}^e_{k+1}) \in \partial F(\boldsymbol{X}^e_{k+1})$, we

can establish the convergence speed as follows:

$$\begin{aligned}
&\min_{i=0,\cdots,K} \|\partial F(\boldsymbol{X}_{i+1}^e)\|_F^2 \leq \frac{\beta^4}{(\alpha+\beta)^2} \cdot \min_{i=0,\cdots,K} \|\boldsymbol{X}_{i,\frac{1}{2}}^e - \boldsymbol{X}_{i+1}^e\|_F^2 \\
&\leq \frac{\beta^4}{(\alpha+\beta)^2} \cdot \frac{1}{K+1} \sum_{k=0}^{K} \|\boldsymbol{X}_{k,\frac{1}{2}}^e - \boldsymbol{X}_{k+1}^e\|_F^2 \\
&\leq \frac{c_0}{K+1}(F(\boldsymbol{X}_0^e) - F^*),
\end{aligned}$$

where $c_0 = \frac{(\beta - L_{\nabla f})\beta^4}{(\alpha+\beta)^2}$.

In conclusion, we finish the proof.

Chapter 5

Hankel Projected Proximal Gradient

5.1 Introduction

In this chapter, we explore the reconstruction of n-size r-spectrally-sparse signals from their m-size subsets, with a particular focus on addressing challenges arising from low sampling density. Such scenarios often lead to slow convergence, posing significant challenges in signal processing. However, the impracticality of complete sampling in many real-world scenarios, due to environmental or economical hardware constraints, necessitates efficient data acquisition strategies. These often involve collecting only a subset of discrete samples, leading to sparse observations, particularly as the signal size increases.

Chapter 4 detailed a proximal gradient-based method tailored for SCS. This LPPG method is noted for its benefits, which include enhanced reconstruction accuracy and consistent convergence towards critical points. However, it encounters computational challenges, particularly under low sampling ratios (denoted as $S_p = m/n$) and in the context of large-scale signal scenarios. These limitations primarily stem from two factors:

1. Increased Number of Iterations: Sparse observations necessitate more iterations for convergence, which is particularly challenging in large-scale signal scenarios.

2. High Computational Complexity per Iteration: The LPPG method exhibits escalated computational demands, especially when the model order is high. This complexity impedes its practicality in handling large-scale datasets efficiently.

This computational burden is obviously undesirable when the noise level is high, as it necessitates a large number of iterations to achieve convergence with a larger Hankel enforcement parameter β. Practically, to trade off the reconstruction accuracy and the computational efficiency, a suitable β is selected for a given noise level and early stopping is applied to terminate the iteration process.

Given these challenges, there is a pressing need for an alternative approach capable of efficiently addressing SSS recovery with reduced computational time, particularly in cases with limited observed entries. While Nesterov's memory techniques, as implemented in the Fast Iterative Shrinkage-Thresholding Algorithm (FISTA) [99], offer a promising avenue with their ability to achieve a fast convergence rate while maintaining the computational simplicity of classical projection gradient algorithms [122, 123], they fall short in nonconvex scenarios. Moreover, the accelerated PG method designed for nonconvex cases [116] only shows marginal improvements in convergence speed.

Therefore, the development of a method that can efficiently and effectively handle the challenges of SSS recovery in sparse and large-scale scenarios remains an open and significant area of research. Such a method would ideally combine the advantages of fast convergence and low computational complexity, making it suitable for a wide range of practical applications in signal processing and related fields.

In the literature, two primary types of acceleration methods are commonly utilized for structured low-rank matrix approximation. These are:

1. **Reduce the Number of Iterations via Second-order Information:** Drawing parallels to Newton's method [124], the proximal Newton-type [106] and Cadzow's Newton-type [59] algorithms are designed with second-order convergence. This approach, however, increases storage and computational complexity. Furthermore, quasi-Newton methods [105, 107, 108],

including limited memory techniques like L-BFGS [125], are also applied. Diagonal or structured approximations of the Hessian matrix are considered in variable metric algorithms [101, 102, 126], facilitating the use of second-order information to decrease the number of iterations required for convergence.

2. **Reduce the Computational Complexity per Iteration via the Intrinsic Structure:** The intrinsic structure of Hankel/Toeplitz matrices [89, 127, 128] can also be leveraged to reduce computational complexity per iteration. This includes the truncated SVD mapping operated on the structured matrix. Additionally, low-rank manifold approaches [57,128,129] are utilized to expedite the SVD process by selecting the tangent space of the low-rank space, thus providing an efficient computational framework.

These methods collectively enhance the efficiency of structured low-rank matrix approximation, contributing significantly to advancements in this domain.

In this work, we introduce the Hankel projected proximal gradient (HPPG) method, designed to optimize both the number of iterations and computational complexity per iteration in spectrally sparse signal recovery. This method significantly accelerates convergence and enhances recovery accuracy by incorporating Hankel projection into the MPG process. The HPPG method's key contributions are as follows:

1. **Efficient Convergence through Hankel Projection:** The integration of Hankel projection within the PG framework facilitates more efficient convergence towards local minima. Unlike traditional line-search strategies [6, 86], HPPG employs an expanded step size, flexibility adjusted in accordance with the sampling ratio. This novel approach ensures accelerated convergence, particularly in scenarios characterized by extremely sparse sampling. Moreover, the Lipschitz constant of the PG is upper bounded by 1, no matter how large the β, this is a significant improvement for noisy cases.

2. **Reduced Computational Complexity:** Alongside a reduction in the number of iterations, the HPPG method significantly decreases the computational complexity per iteration. For n-size r-spectrally-sparse signals, the complexity is reduced from $\mathcal{O}(r^4 n +$

$r^3 n \log n)$ to $\mathcal{O}(r^2 n + rn \log n)$, retaining the Hankel structure. Simulation results verifies the effectiveness of the HPPG method in terms of both accuracy and computational speed.

These contributions collectively demonstrate the HPPG method's potential in providing a more efficient approach to the recovery of SSSs, especially in the context of sparse sampling.

5.2 The Optimization Formulation

Similar to Chapter 4, this work focuses on the recovery of one-dimensional SSS, with extensions to multilevel case simulations included in Section 5.5. For clarity of exposition, we consider the example of undamped SSS.

$$\boldsymbol{x} = \sum_{k=1}^{r} d_k \boldsymbol{y}(f_k;\ n), \tag{5.1}$$

where d_k denotes the complex coefficient, and $\boldsymbol{y}(f_k; n)$ encapsulates the frequency of the kth spectral component.

The task at hand is to recover the SSS $\boldsymbol{x}$ from its partial observations $\boldsymbol{s}$:

$$\text{find} \quad \boldsymbol{x} \quad \text{s.t.} \quad \mathcal{P}_\Omega \boldsymbol{x} = \boldsymbol{s}. \tag{5.2}$$

The Hankel matrix $\mathcal{H}\boldsymbol{x}$ can be decomposed using a Vandermonde structure:

$$\mathcal{H}\boldsymbol{x} = \sum_{k=1}^{r} d_k \boldsymbol{y}(f_k;\ p)\boldsymbol{y}(f_k;\ q)^{\mathsf{T}} \in \mathbb{C}^{p \times q}. \tag{5.3}$$

Typically, as described in [2, 56], we choose $p \approx q \approx n/2$ $(p \leq q)$ to make $\mathcal{H}\boldsymbol{x}$ approach a square matrix form, which enhances resolution performance. Additionally, we demonstrate that this squared enhanced matrix offers a better Lipschitz constant and consequently an improved condition number of the Hessian matrix for the continuous part of the objective function.

Remark 13. *To mitigate performance degradation, researchers have incorporated prior knowledge of the target signal [53] and employed the Hankel-Toeplitz model for undamped SSS [44]. However, these methods come with their own set of model assumptions and limitations, particularly in handling the damping factor.*

The rank-constrained nonconvex problem is thus formulated as:

$$\min_{\boldsymbol{X}^e,\boldsymbol{x}} \ \delta(\operatorname{rank}(\boldsymbol{X}^e) \leq r) + \frac{1}{2}\|\boldsymbol{s} - \mathcal{P}_\Omega \boldsymbol{x}\|_2^2 + \frac{\beta}{2}\|\boldsymbol{X}^e - \mathcal{H}\boldsymbol{x}\|_F^2. \tag{5.4}$$

In contrast to the objective function (4.6) addressed by LPPG, this simplified formulation circumvents the need for its regularization term.

5.3 A Modified PG Method With Single Variable

As demonstrated in Chapter 4, the direct application of PG [6] to the SSS recovery in array norm resulted in unacceptably slow convergence due to the small step size. Specifically, the standard PG applied directly to (5.4) yields $L_{\nabla f} = 1 + \beta \min(p, q)$[1], leading to a small value for γ in the proximal gradient descent process. To enhance the step size, a reformulation of (5.4), focusing solely on the single variable $\boldsymbol{X}$, is presented in [86]:

$$F(\boldsymbol{X}^e) \coloneqq f(\boldsymbol{X}^e) + g(\boldsymbol{X}^e), \tag{5.5}$$

$$f(\boldsymbol{X}^e) \coloneqq \min_{\boldsymbol{x}} \ \frac{1}{2}\|\boldsymbol{s} - \mathcal{P}_\Omega \boldsymbol{x}\|^2 + \frac{\beta}{2}\|\boldsymbol{X}^e - \mathcal{H}\boldsymbol{x}\|_F^2, \tag{5.6}$$

$$g(\boldsymbol{X}^e) \coloneqq \delta(\operatorname{rank}(\boldsymbol{X}^e) \leq r). \tag{5.7}$$

Supposing the Lipschitz constant of $\nabla f(\boldsymbol{X}^e)$ is $L_{\nabla f}$, we have $L_{\nabla f} \leq \beta$, irrespective of the dimensionality, as shown in Proposition 4.

[1]This Lipschitz constant partly reveals the reason for the squared enhanced Hankel matrix.

Proposition 4 (Proposition 2 in [86]). *We have*

$$\nabla f(\boldsymbol{X}^e) = -\beta\mathcal{H}\mathcal{L}\mathcal{P}_\Omega \boldsymbol{s} - \beta^2\mathcal{H}\mathcal{L}\mathcal{H}^*\boldsymbol{X}^e + \beta\boldsymbol{X}^e,$$

and $\nabla^2 f(\boldsymbol{X}^e) = \beta(\mathcal{I} - \beta\mathcal{H}\mathcal{L}\mathcal{H}^*)$ *with* $\mathcal{L} = (\mathcal{P}_\Omega + \beta\mathcal{H}^*\mathcal{H})^{-1}$. *Moreover, we have* $L_{\nabla f}$ *as* $0 < L_{\nabla f} \leq \beta$.

Additionally, the equation (5.6) represents a least-square problem with a closed-form solution given by

$$\boldsymbol{x}^\star = \mathcal{L} \cdot (\mathcal{P}_\Omega \boldsymbol{s} + \beta\mathcal{H}^*\boldsymbol{X}^e). \tag{5.8}$$

Note that $\mathcal{L} = (\mathcal{P}_\Omega + \beta\mathcal{H}^*\mathcal{H})^{-1}$ is a diagonal matrix with positive entries, which can be efficiently evaluated. The modified proximal gradient then takes the form:

$$\operatorname{prox}_{\gamma g}\left(\boldsymbol{X}_k^e - \gamma\nabla f(\boldsymbol{X}_k^e)\right) = \mathcal{T}_r\left(\boldsymbol{X}_k^e - \gamma\nabla f(\boldsymbol{X}_k^e)\right), \tag{5.9}$$

where $\mathcal{T}_r(\cdot)$ denotes the operator that truncates the matrix to its top r singular values.

As a result, the step size γ of the modified PG method should satisfy $\gamma \leq \frac{1}{\beta}$, which is dimension-independent. However, the approaches outlined in [86] are less effective at lower sampling rates due to the following reason: the expression $\frac{1}{2}\|\boldsymbol{s} - \mathcal{P}_\Omega\boldsymbol{x}\|_2^2$ within the modified objective function (5.4) is typically small, which consequently leads to a relatively larger value of β. Moreover, when the noise level is large it is troublesome to adjust the Hankel enforcement parameter β between the reconstruction accuracy and the computational efficiency, especially in the case of high computational complexity per iteration.

5.4 The Modified PG Method in Hankel Space

In this section, we employ a Hankel space projection denoted as $\mathcal{HS}$ before applying the proximal mapping. This approach offers two significant advantages: 1) a more accurate estimation of $L_{\nabla f}$, which is related to the sampling rate; and 2) a faster proximal mapping due to the Hankel

structure.

5.4.1 A Sampling-rate-related $\hat{L}_{\nabla f}$ in Hankel Space

Definition 5.4.1 (Hankel matrix projection on low-rank matrix)**.** *For a rank-r matrix* $\boldsymbol{X}^e \in \mathbb{C}^{n_1 \times n_2}$ *that admits a SVD as* $\boldsymbol{X}^e = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathsf{H}}$*, the Hankel matrix projection is defined as* $\mathcal{HS}\boldsymbol{X}^e = \mathcal{H}\mathcal{W}^{-1}\mathcal{H}^*\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathsf{H}}$*.*

As indicated in Prop. 4, we have $\nabla^2 f = \beta(\mathcal{I} - \beta\mathcal{H}\mathcal{L}\mathcal{H}^*)$. If $\boldsymbol{X}^e$ is projected into the Hankel space as $\hat{\boldsymbol{X}}^e = \mathcal{HS}\boldsymbol{X}^e = \mathcal{H}\hat{\boldsymbol{x}}$, the operator norm of $\nabla^2 f$ on Hankel space is calculated as:

$$\sup_{\|\mathcal{H}\hat{\boldsymbol{x}}\|_F=1} \|\beta(\mathcal{I} - \beta\mathcal{H}\mathcal{L}\mathcal{H}^*)\mathcal{H}\hat{\boldsymbol{x}}\|_F \tag{5.10}$$

$$= \sup_{\|\mathcal{H}\hat{\boldsymbol{x}}\|_F=1} \beta\|(\mathcal{I} - \beta\mathcal{H}(\mathcal{P}_\Omega^*\mathcal{P}_\Omega + \beta\mathcal{H}^*\mathcal{H})^{-1}\mathcal{H}^*)\mathcal{H}\hat{\boldsymbol{x}}\|_F \tag{5.11}$$

$$= \sup_{\|\mathcal{H}\hat{\boldsymbol{x}}\|_F=1} \beta\|\mathcal{H}(\mathcal{I} - \beta(\mathcal{P}_\Omega^*\mathcal{P}_\Omega + \beta\mathcal{H}^*\mathcal{H})^{-1}\mathcal{H}^*\mathcal{H})\hat{\boldsymbol{x}}\|_F. \tag{5.12}$$

Given $\mathcal{D} = (\mathcal{I} - \beta(\mathcal{P}_\Omega^*\mathcal{P}_\Omega + \beta\mathcal{H}^*\mathcal{H})^{-1}\mathcal{H}^*\mathcal{H}) \in \mathbb{C}^{n\times n}$, we observe that $\mathcal{D}$ is a diagonal matrix with elements

$$\mathcal{D}_{ii} = \begin{cases} \frac{1}{1+\beta w_i} & \text{if } i \in \Omega \\ 0 & \text{if } i \notin \Omega \end{cases}, \quad \text{for} \quad i = 1, 2, \cdots, n,$$

where w_i is the i-th element of $\boldsymbol{w}$. Therefore, the operator norm of $\nabla^2 f$ equals $\beta\|\mathcal{H}\mathcal{D}\hat{\boldsymbol{x}}\|_F$, which is bounded by $\frac{\beta}{1+\beta}\|\mathcal{H}\mathcal{P}_\Omega\hat{\boldsymbol{x}}\|_F$.

Considering the sampling ratio of the sampling operator $\mathcal{P}_\Omega$ is S_p, we can derive an improved approximation of $L_{\nabla f}$ as

$$\hat{L}_{\nabla f} = \frac{\beta S_p}{1+\beta}. \tag{5.13}$$

This $\hat{L}_{\nabla f}$ demonstrates two notable characteristics:

1. Its magnitude is proportional to S_p.

2. It is bounded by 1, regardless of the magnitude of β.

Similarly, the operator norm of the inverse of the Hessian operator norm of $\nabla^2 f^{-1}$ can be approximated as $\frac{1+\beta \max(p,q)}{\beta}$[2]. Consequently, the condition number κ, which determines the convergence speed for the gradient descent method, is given by

$$\kappa(\nabla^2 f) \approx \frac{\beta S_p}{1+\beta} \cdot \frac{1+\beta \max(p,q)}{\beta} = \frac{S_p(1+\beta \max(p,q))}{1+\beta}. \tag{5.14}$$

The $\kappa(\nabla^2 f)$ is reduced from $1+\beta \max(p,q)$, which also explains the preference for an enhanced Hankel matrix close to a square matrix.

Remark 14. *It is important to note that this $\hat{L}_{\nabla f}$ based on Hankel space significantly differs from a line-search strategy. Since β acts as a precise and tight upper bound for the original $L_{\nabla f}$, the step size returned by the line-search mechanism is typically $1/\beta$. The advantage of HPPG over the line-search approach is evident in the numerical results presented later. Moreover, the HPPG method is more efficient to choose a larger β which is often the troublesome procedure in noisy scenarios.*

5.4.2 The HPPG Algorithm With Convergence

To establish a convergence guarantee for the HPPG, equipped with the optimized $\hat{L}_{\nabla f}$, we introduce the following condition, denoted as C, to guide the update of $\boldsymbol{X}^e_{k+1}$:

$$\text{C: } \left\langle \frac{1}{\hat{\gamma}} \nabla f(\hat{\boldsymbol{X}}_k) - \frac{1}{\gamma} \nabla f(\boldsymbol{X}^e_k), \hat{\boldsymbol{X}}_k - \boldsymbol{X}^e_k \right\rangle \geq 0. \tag{5.15}$$

Eq. (5.15) is set to ensure the monotone decrease of the HPPG, thus guaranteeing the sufficient descent property as outlined in [116]. Furthermore, when condition C holds true, each iteration necessitates only a single proximal mapping operation. The HPPG algorithm is detailed in Algorithm 7.

[2]The inverse of the Hessian operator is not well-defined due to the potential existence of zero eigenvalues.

Algorithm 7 HPPG for SCS

1: **Input:** The observations: $\boldsymbol{s}$,
2: Hankel enforcement parameter: β
3: $\boldsymbol{X}_0^e \leftarrow \text{prox}_{\gamma g}(\mathcal{H}\boldsymbol{s})$, constant $\gamma = \frac{1}{\beta}, \hat{\gamma} = \frac{1+\beta}{\beta S_p}$
4: **while** $k < K$ **or** $\|(\boldsymbol{X}_k^e - \boldsymbol{X}_{k-1}^e)\|_F \geq \epsilon \|\boldsymbol{X}_{k-1}^e\|_F$ **do**
5: $\hat{\boldsymbol{X}}^e{}_k \leftarrow \mathcal{H}\mathcal{S}\boldsymbol{X}_k^e$
6: $\boldsymbol{Z}_{k+1} \leftarrow \text{prox}_{\hat{\gamma} g}\left(\hat{\boldsymbol{X}}^e{}_k - \hat{\gamma}\nabla f(\hat{\boldsymbol{X}}^e{}_k)\right)$
7: **if** Update Condition C **then**
8: $\boldsymbol{X}_{k+1}^e \leftarrow \boldsymbol{Z}_{k+1}$
9: **else**
10: $\boldsymbol{X}_{k+1}^e \leftarrow \text{prox}_{\gamma g}\left(\boldsymbol{X}_k^e - \gamma \nabla f(\boldsymbol{X}_k^e)\right)$
11: **end if**
12: **end while**
13: **Output:** $\boldsymbol{x}^\dagger \leftarrow \mathcal{L}(\mathcal{P}_\Omega^* \boldsymbol{s} + \beta \mathcal{H}^* \boldsymbol{X}_k^e)$, $\boldsymbol{X}_k^e$ in the last iteration

When the update condition is met, we can show that $F(\boldsymbol{Z}_{k+1}) \leq F(\boldsymbol{X}_{k+1}^e)$, where $\boldsymbol{Z}_{k+1}$ is determined by the proposed Hankel matrix proximal step (cf. Line 3, Algorithm 2), and $\boldsymbol{X}_{k+1}^e$ is obtained from the modified PG step (cf. Line 7, Algorithm 2). Therefore, the proposed HPPG method exhibits the Sufficient Descent Property as demonstrated in Chapter 4. Consequently, similar to the proof of LPPG, the convergence of the HPPG method to critical points is guaranteed as $0 \in \partial F(\boldsymbol{X}^{e*})$, where $\boldsymbol{X}^{e*}$ is the accumulation point of the sequence $\{\boldsymbol{X}_k^e\}_{k=1,2,\ldots}$ [116].

Remark 15. *The HPPG outlined in Algorithm 7 is a monotone decreasing iterative process converging to the critical points. However, in cases of ill-conditioned problems, a monotone algorithm may exhibit slow convergence due to short step sizes or zigzagging trajectories while maintaining a non-increasing objective function [130]. The Update Condition C in Algorithm 7 is a conservative strategy to ensure a strictly decreasing property. It is possible to allow for occasional increases in the objective function value to improve convergence [131–134], although this nonmonotone version of HPPG is not detailed here for brevity.*

5.4.3 Computational Complexity per Iteration

Besides the enhanced fast convergence due to the larger step size $\hat{\gamma} \leq \frac{1}{\hat{L}_{\nabla f}}$, the computational complexity per iteration of the HPPG method is significantly reduced to $\mathcal{O}(r^2 n + rn \log n)$ from

the traditional $\mathcal{O}(n^3)$ complexity.

The proximal mapping on the function $\delta(\text{rank}(\boldsymbol{X}^e) \leq r)$ acts as a truncated singular value decomposition (SVD) operator [118]. In the modified PG step (cf. Line 7, Algorithm 2), the truncated SVD is performed on a linear combination of a Hankel matrix and a low-rank matrix. This process can be accelerated using the Lanczos method and FFT [51], resulting in a computational complexity of $\mathcal{O}(rn \log n + r^2 n)$ flops. In contrast, the proposed Hankel matrix proximal step (cf. Line 3, Algorithm 2), which operates directly on the Hankel matrix, requires only $\mathcal{O}(rn \log n)$ flops.

Furthermore, the Hankel projection on a low-rank matrix can be computed efficiently using fast convolution techniques, incurring a computational cost of $\mathcal{O}(r^2 n + rn \log n)$ flops [51]. The update condition in (5.15) involves calculating an inner product, which also has a computational complexity of $\mathcal{O}(r^2 n + rn \log n)$ flops. Therefore, the total computational complexity per iteration for the HPPG method remains at $\mathcal{O}(r^2 n + rn \log n)$, making it a highly efficient approach for spectrally sparse signal recovery.

5.5 Simulations

Remark 16. *Unlike the LPPG discussed in Chapter 4, the convergence of the HPPG is determined by the difference between consecutive iterations. This approach is motivated by three reasons: (1) the boundedness of the subgradient of the PG-based method by the adjacent difference, (2) the computational efficiency of using the adjacent difference for benchmark methods, and (3) the variability of the Hankel enforcement parameter β, which renders the accuracy of the subgradient for early termination.*

To evaluate the recovery capability of the HPPG method, we consider a successful recovery when the NMSE is below 10^{-3}. The HPPG method, characterized by its monotone decreasing property and larger step size, shows superior performance in finding favorable critical points for the nonconvex formulation. This is empirically supported by the recovery phase transition curve

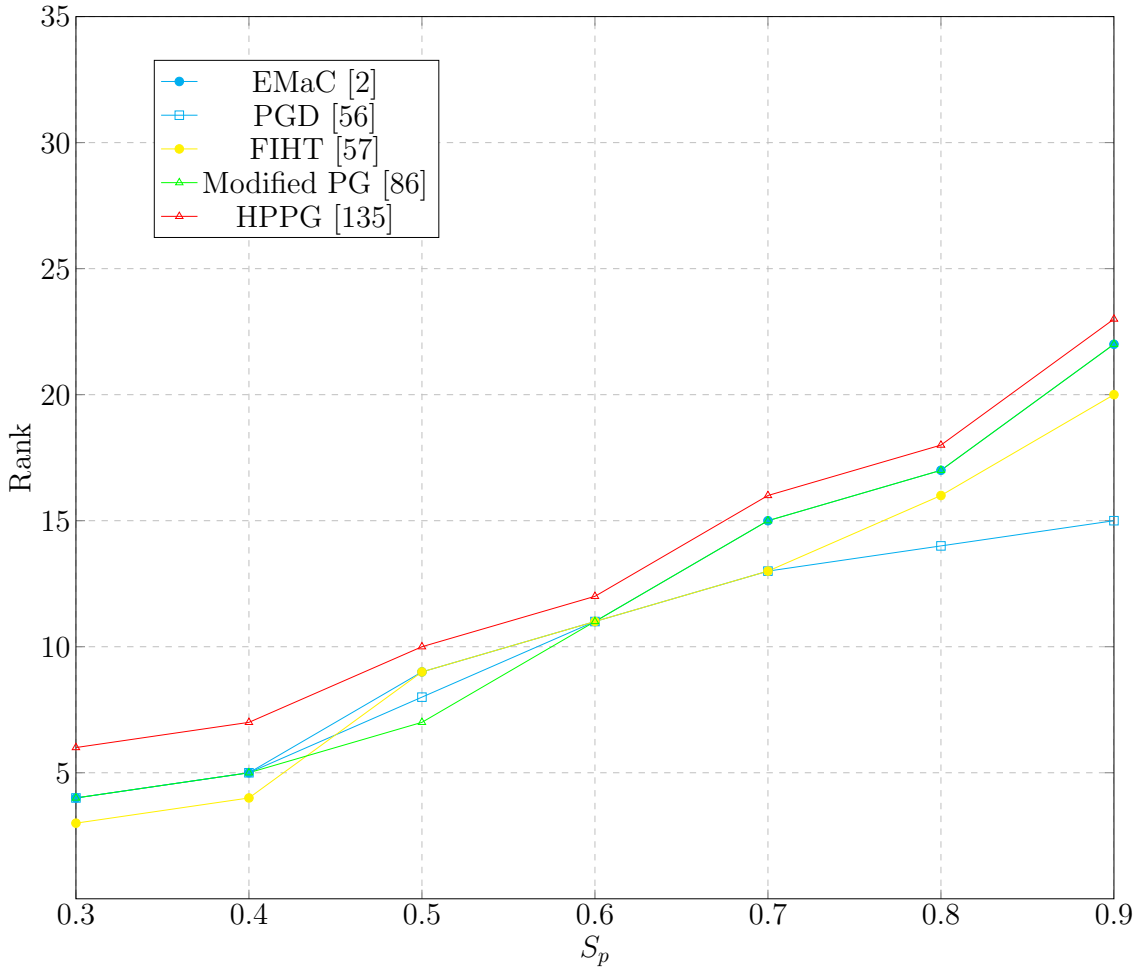


Figure 5.1: Phase transition curve $n = 63$, $(n_1, n_2) = (32, 32)$.

depicted in Fig. 1. The curve represents the threshold above which the tested methods exhibit a 50% failure rate. The results demonstrate that the proposed HPPG algorithm outperforms other methods under consideration.

We then assess the rapid convergence of the HPPG method with respect to NMSE, the number of iterations, and computational time. This assessment is particularly relevant for challenging sampling ratios, specifically $p = \{0.2,\ 0.1\}$. The results, as summarized in Table 5.1, demonstrate that HPPG achieves the fastest convergence among the methods tested. This notable performance is primarily due to its reduced count of iterations and the efficient computational complexity per iteration. HPPG's ability to achieve rapid convergence in scenarios with low sampling ratios is a significant advantage, emphasizing its effectiveness in spectrally sparse signal recovery.

Table 5.1: NMSEs and time(s) for different p: $\boldsymbol{x} \in \mathbb{C}^{65\times 65}$, the resulting $\boldsymbol{X}^e \in \mathbb{C}^{1089\times 1089}$, and $r = 10$.

Sampling rate	p=0.2			p=0.1		
Criterion	NMSE	Iter	Time(s)	NMSE	Iter	Time(s)
EMaC [2]	2.1e-6	86	3.9	3.4e-6	183	6.9
PGD [56]	7.2e-6	58	2.1	8.6e-6	76	2.8
FIHT [57]	2.02e-7	22	1.8	3.6e-7	46	2.6
Modified PG [86]	5.6e-6	80	4.2	1.5e-5	159	8.1
HPPG [135]	4.4e-7	15 (15)[3]	1.1	4.4e-7	34 (34)	1.6

Finally, we investigate the robustness of the HPPG method in scenarios involving additive zero-mean Gaussian noise. As outlined in [86], a larger Hankel enforcement parameter (β) is often associated with more accurate recovery in the presence of noise. The HPPG method, with its advantageously upper-bounded $\hat{L}_{\nabla f}$, demonstrates exceptional effectiveness, especially for relatively large values of β. This results in enhanced reconstruction accuracy. The performance of HPPG, as compared to benchmark methods, is illustrated in Fig. 2. Across a range of signal-to-noise ratios (SNR), HPPG consistently outperforms the other methods considered in the study, underscoring its robustness in noisy environments.

5.6 Conclusions

In this chapter, we introduce the HPPG method, designed for the reconstruction of spectrally sparse signals, particularly under very low sampling ratios. The HPPG approach effectively addresses and mitigates the high computational demands associated with the LPPG method. Notably, HPPG reduces both the number of iterations required for convergence and the computational complexity inherent in each iteration.

The HPPG method is distinguished by its ability to utilize a larger step size in the modified proximal gradient process. This step size is directly related to the sampling rate and is upper bounded, independent of the Hankel enforcement parameter. Additionally, the HPPG method leverages the Hankel structure to further enhance its computational efficiency.

[3]The number in brackets indicates the iterations when *Condition C* is updated.

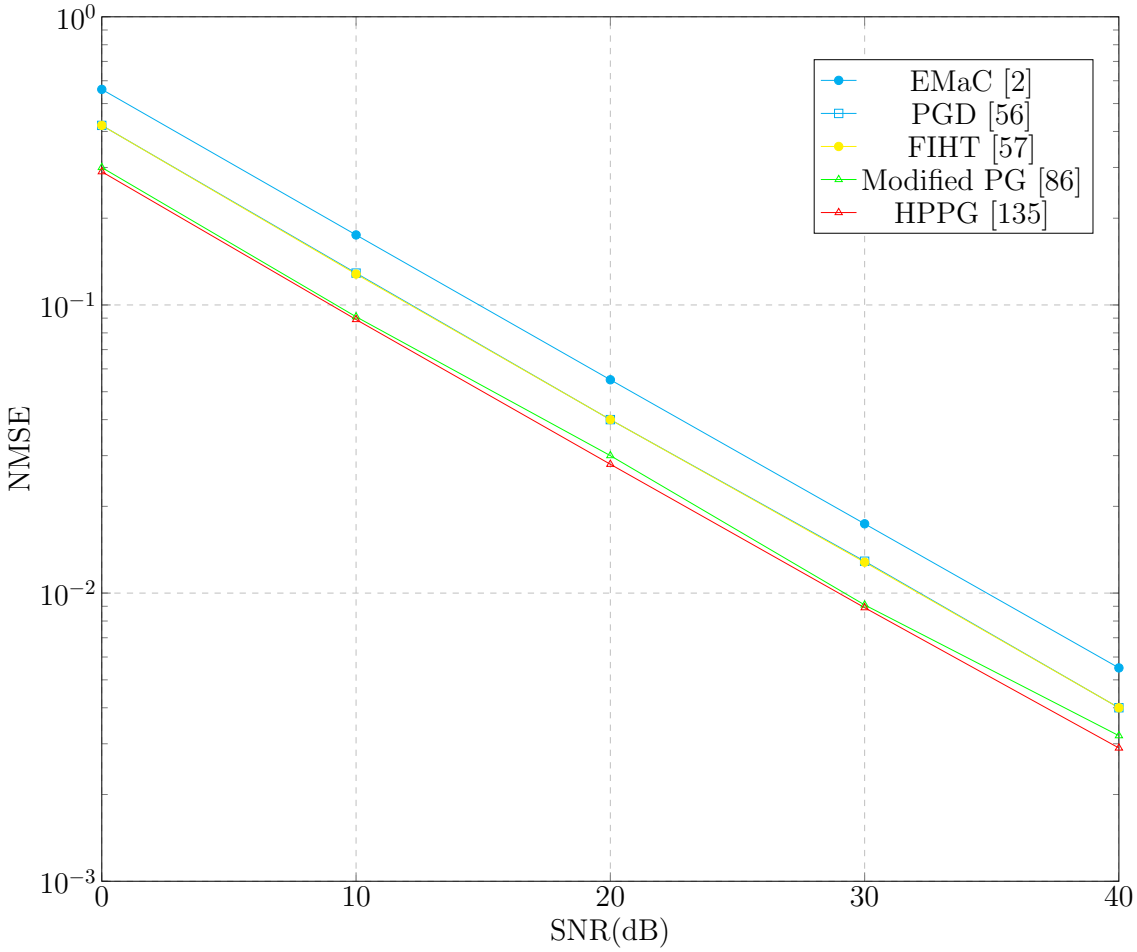


Figure 5.2: Performance under additive noise: $\boldsymbol{X} \in \mathbb{C}^{11\times 11\times 11}$, the resulting $\boldsymbol{X}^e \in \mathbb{C}^{216\times 216}$, $p = 0.2$, and $r = 5$.

Empirical evaluations of the HPPG method demonstrate its superior performance compared to existing state-of-the-art approaches. The advantages are particularly evident in terms of recovery accuracy and computational speed, making HPPG a highly effective and efficient tool for the reconstruction of spectrally sparse signals in scenarios characterized by low sampling rates. This chapter details the formulation, implementation, and empirical validation of the HPPG method, underscoring its significant contributions to the field of signal processing.

Chapter 6

Hessian Proximal Gradient Methods

The LPPG method, introduced in Chapter 4, effectively recovers SSSs of medium size with considerable accuracy. The HPPG method, detailed in Chapter 5, addresses computational complexity challenges by adopting an increased step size and the utilization of the Hankel structure. This method overcomes low-sampling issues and bounds the Lipschitz constant of the smooth component at 1, regardless of the magnitude of the Hankel enforcement parameter β. However, akin to other first-order methods, HPPG suffers from slow convergence when β is substantially high, as needed in high noise scenarios. This slow convergence is due to the condition number, $\kappa(\nabla^2 f)$, being proportional to β. Typically, achieving fast convergence requires compromising reconstruction accuracy, often managed by employing a smaller regularization coefficient for the Hankel structure constraint.

In addition to the utilization of intrinsic structures, we present an accelerated PG algorithm incorporating Hessian information. To achieve both rapid convergence and accurate reconstruction, we introduce the Hessian proximal gradient (HPG) method. This approach selects a larger regularization coefficient without compromising reconstruction accuracy. Remarkably, the computational complexity of the forward step with a preconditioned Hessian is reduced from $\mathcal{O}(n^3)$ to $\mathcal{O}(r^2 n + rn \log n)$ for n-size r-spectrally-sparse signals. Our numerical simulations demonstrate that HPG surpasses state-of-the-art methods in computational speed and reconstruction accuracy, especially under high noise levels.

The objective function for HPG remains the same single-variable formulation as presented in the previous chapter:

$$F(\boldsymbol{X}^e) \coloneqq f(\boldsymbol{X}^e) + g(\boldsymbol{X}^e), \tag{6.1}$$

$$f(\boldsymbol{X}^e) \coloneqq \min_{\boldsymbol{x}} \ \frac{1}{2}\|\boldsymbol{s} - \mathcal{P}_\Omega \boldsymbol{x}\|^2 + \frac{\beta}{2}\|\boldsymbol{X}^e - \mathcal{H}\boldsymbol{x}\|_F^2, \tag{6.2}$$

$$g(\boldsymbol{X}^e) \coloneqq \delta(\text{rank}(\boldsymbol{X}^e) \leq r). \tag{6.3}$$

HPG is a second-order proximal method based on this modified single-variable formulation. The following sections provide a brief overview of existing literature regarding proximal methods, leading to a detailed exposition of our HPG method.

6.1 Introduction

Firstly, we state the definition of the scaled proximal mapping, which is the key to the proximal Newton method.

Definition 6.1.1 (Scaled Proximal Mapping: $\text{prox}_g^{\boldsymbol{H}}(\boldsymbol{X})$)**.** *The scaled proximal mapping is defined as the optimization problem:*

$$prox_g^{\boldsymbol{H}}(\boldsymbol{X}) \coloneqq \arg\min_{\boldsymbol{Z}} \frac{1}{2}\|\boldsymbol{X} - \boldsymbol{Z}\|_{\boldsymbol{H}} + g(\boldsymbol{Z}), \tag{6.4}$$

where $\|\boldsymbol{X}\|_{\boldsymbol{H}} = trace(\boldsymbol{X}^\top \boldsymbol{H}\boldsymbol{X})$ *and* $\boldsymbol{H} \succeq 0$*. When* $\boldsymbol{H} = \boldsymbol{I}$*, this reduces to the standard proximal mapping.*

The key part of proximal methods lies in solving a subproblem of the form

$$\arg\min_{\boldsymbol{D}_k} \left[f(\boldsymbol{X}_k) + \nabla f(\boldsymbol{X}_k)^\top \boldsymbol{D}_k + \frac{1}{2}\boldsymbol{D}_k^\top \boldsymbol{H}_k \boldsymbol{D}_k + g(\boldsymbol{X}_k + \boldsymbol{D}_k) \right] \tag{6.5}$$

at each iteration, where $\boldsymbol{H}_k$ is typically a positive definite matrix. The solution $\boldsymbol{D}_k$ in this

context is unique and is characterized using the proximity operator:

$$\boldsymbol{D}_k \coloneqq \operatorname{prox}_g^{\boldsymbol{H}_k}\left(\boldsymbol{X}_k - \boldsymbol{H}_k^{-1}\nabla f(\boldsymbol{X}_k)\right), \tag{6.6}$$

$$\operatorname{prox}_g^{\boldsymbol{H}_k}(\boldsymbol{X}) \coloneqq \arg\min_{\boldsymbol{Z}} \frac{1}{2}\|\boldsymbol{X}-\boldsymbol{Z}\|_{\boldsymbol{H}_k} + g(\boldsymbol{Z}). \tag{6.7}$$

Subsequently, the next iteration is defined as $\boldsymbol{X}_{k+1} = \boldsymbol{X}_k + \boldsymbol{D}_k$.

First-order methods select $\boldsymbol{H}$ as a positive multiple of the identity matrix, i.e., $\boldsymbol{H} = \gamma^{-1}\boldsymbol{I}$, where $\gamma > 0$. Due to their computational efficiency in many scenarios, or even analytical solvability, these methods have been a focus of intensive research over the past decade. They encompass numerous variants, including Nesterov's acceleration techniques [99, 136]. Known for their ease of implementation and well-documented convergence properties, first-order methods, however, are often criticized for their poor convergence rates, particularly when the condition number of $\boldsymbol{H}$ is significantly large. This can result in a large number of iterations to achieve an acceptable solution approximation.

Variable metric proximal methods allow for the alteration of $\boldsymbol{H}$ in each iteration based on specific criteria [126, 137, 138]. However, these methods often lose the analytical solvability of the subproblem (6.5), necessitating further investigation into the solutions of these subproblems. This typically includes an inexactness criterion [139, 140] and considerations for structured or diagonal $\boldsymbol{H}$ [126].

Second-order methods, closely related to variable metric approaches, incorporate second-order information of f in $\boldsymbol{H}_k$, approximating $\boldsymbol{H}_k \approx \nabla^2 f(\boldsymbol{X}_k)$. Proximal Newton methods [7, 141], for instance, directly use $\boldsymbol{H}_k = \nabla^2 f(\boldsymbol{X}_k)$ and are acknowledged for their excellent convergence properties akin to the Newton method for smooth unconstrained minimization [7]. Conversely, due to the computational expense associated with such approaches, particularly in high-dimensional problems, proximal quasi-Newton or proximal Newton-type methods are often favored [105, 142]. These methods involve approximating the Hessian using quasi-Newton or limited memory quasi-Newton strategies. The practical challenge lies in the exact solution of the proximal Newton step, which requires the numerical resolution of problem (6.5).

In this chapter, we introduce the HPG method, a novel approach that addresses the challenges of slow convergence and limited recovery accuracy in SSS recovery. This method stands out by incorporating second-order information, leading to two major contributions: Firstly, the utilization of a preconditioned Hessian matrix significantly accelerates convergence towards local minima. This efficiency is achieved with fewer iterations, even across a broad spectrum of regularization coefficients for the Hankel structure constraint. Secondly, we achieve a substantial reduction in computational complexity per iteration, from $\mathcal{O}(n^3)$ to $\mathcal{O}(r^2 n + rn \log n)$, for n-size r-spectrally-sparse signals. This reduction is accomplished by leveraging the Hankel and low-rank structure inherent in these signals. To our knowledge, this represents the lowest computational complexity achieved in a Hessian-based method for this class of problems. Numerical simulations underscore the significant advancements of the HPG method over existing benchmarks, showcasing marked improvements in both accuracy and computational speed.

6.2 The Proximal Newton Method

Algorithm 8 Proximal Newton Method [7]

1: Set $\boldsymbol{H}_k \coloneqq \nabla^2 g(\boldsymbol{X}_k)$.
2: **Initialize** $\boldsymbol{X}_0$.
3: **while** condition or $k = 0, 1, 2, \ldots$ **do**
4: Update $\boldsymbol{X}_{k+1} \leftarrow \operatorname{prox}_g^{\boldsymbol{H}_k}\left(\boldsymbol{X}_k - \boldsymbol{H}_k^{-1}\nabla f(\boldsymbol{X}_k)\right)$.
5: **end while**

Although the proximal Newton method can converge under some mild assumptions [7], such as the convexity of f [105, 142] or global continuity of ∇f [106, 143], there are still computational bottlenecks to consider. Notably, some studies focusing on the ℓ_1-norm ($g = \|\cdot\|_{\ell_1}$) have also been investigated [144, 145]. However, despite its improved convergence rate, the proximal Newton method faces two significant computational challenges:

1. For a dense $\boldsymbol{H}_k$, calculating the inverse $\boldsymbol{H}_k^{-1}$ is often not feasible.

2. The scaled proximal mapping cannot be efficiently solved with a standard inner solver, except for ℓ_1 regularized linear models.

To address these issues, the Hankel structure is employed in the HPG method for solving SCS problems.

6.3 The HPG Algorithm with Convergence

In this section, we introduce the HPG method, a second-order proximal approach based on the single-variable formulation presented in equations (6.1)–(6.3). The HPG method, detailed in Algorithm 9, aims to accelerate convergence by incorporating Hessian information. Key to this method is the introduction of the preconditioned Hessian matrix $\tilde{\boldsymbol{H}}_k$, which supersedes the original Hessian matrix $\boldsymbol{H}_k$ in the proximal Newton step. The Hessian matrix, denoted as $\boldsymbol{H}_k \coloneqq \nabla^2 f(\boldsymbol{X}_k^e)$, captures the local geometry of $f(\boldsymbol{X}_k^e)$ at the k-th iteration, facilitating a more efficient progression toward stationary points. Traditionally, the inverse of the Hessian matrix is employed for acceleration, albeit at a significant computational cost of $\mathcal{O}(n^3)$ [146]. Our proposed algorithm leverages Hessian information while maintaining a more manageable computational complexity of $\mathcal{O}(r^2 n + rn \log n)$ per iteration, as established in Proposition 5.

To address specific challenges in the HPG method, we introduce two key modifications. Firstly, due to the potential presence of zero eigenvalues in the Hessian matrix $\boldsymbol{H}_k$, which impede its invertibility, we propose the use of a *preconditioned Hessian* matrix $\tilde{\boldsymbol{H}}_k$. This matrix is defined as follows:

$$\tilde{\boldsymbol{H}}_k \coloneqq \beta(\mathcal{I} - \beta\mathcal{H}\tilde{\mathcal{L}}\mathcal{H}^*) \succ 0, \quad \tilde{\mathcal{L}} \coloneqq (\mathcal{I} + \beta\mathcal{H}^*\mathcal{H})^{-1}.$$

Secondly, to ensure the convergence of the HPG method, which is implemented using $\tilde{\boldsymbol{H}}_k$, we introduce the following update criterion:

$$\text{C: } \langle \frac{1}{\gamma}\tilde{\boldsymbol{H}}_k{}^{-1}\nabla f(\boldsymbol{X}_k^e) - \nabla f(\boldsymbol{X}_k^e),\ \boldsymbol{Z}_{k+1} - \boldsymbol{X}_k^e \rangle \geq 0, \tag{6.8}$$

where $\boldsymbol{Z}_{k+1}$ is derived by applying the proximal mapping with the preconditioned Hessian $\tilde{\boldsymbol{H}}_k$. In cases where condition (6.8) is not satisfied, the standard PG method with a guaranteed sufficient descent property is employed [116]. The HPG method, incorporating these modifications, is

detailed in Algorithm 9.

Algorithm 9 HPG for SCS

1: Set $\boldsymbol{X}_0^e \leftarrow \text{prox}_{\gamma g}\mathcal{H}\boldsymbol{s}$, constant $\gamma \leftarrow \frac{1}{\beta}$.
2: **while** $k < K$ **or** $\|(\boldsymbol{X}_k^e - \boldsymbol{X}_{k-1}^e)\|_F \geq \epsilon\|\boldsymbol{X}_{k-1}^e\|_F$ **do**
3: $\quad \boldsymbol{Z}_{k+1} \leftarrow \text{prox}_{\gamma g}\left(\boldsymbol{X}_k^e - \tilde{\boldsymbol{H}}_k^{-1}\nabla f(\boldsymbol{X}_k^e)\right)$;
4: $\quad$ **if** $\langle \frac{1}{\gamma}\tilde{\boldsymbol{H}}_k^{-1}\nabla f(\boldsymbol{X}_k^e) - \nabla f(\boldsymbol{X}_k^e), \boldsymbol{Z}_{k+1} - \boldsymbol{X}_k^e\rangle \geq 0$ **then**
5: $\quad\quad \boldsymbol{X}_{k+1}^e \leftarrow \boldsymbol{Z}_{k+1}$;
6: $\quad$ **else**
7: $\quad\quad \boldsymbol{X}_{k+1}^e \leftarrow \text{prox}_{\gamma g}\left(\boldsymbol{X}_k^e - \gamma\nabla f(\boldsymbol{X}_k^e)\right)$;
8: $\quad$ **end if**
9: **end while**
Output: $\boldsymbol{X}_k^e$ in the last iteration, $\boldsymbol{x}^\dagger \leftarrow \mathcal{L}(\mathcal{P}_\Omega^*\boldsymbol{s} + \beta\mathcal{H}^*\boldsymbol{X}_k^e)$

Remark 17. *Analogous to the HPPG, the HPG can be adapted with a soft condition C, enhancing its robustness by permitting an increase in the objective function when necessary.*

When the update criterion is satisfied, it can be demonstrated that the Hessian proximal step (as outlined in Line 2 of Algorithm 9) achieves a greater descent than the standard PG step (refer to Line 6, Algorithm 2). Consequently, the HPG method also fulfills the Sufficient Descent Property as detailed in [116]. Thus, convergence of the HPG method to critical points is assured, satisfying $0 \in \partial F(\boldsymbol{X}^{e*})$. Here, $\boldsymbol{X}^{e*}$ denotes the accumulation point of the sequence $\{\boldsymbol{X}_k^e\}_{k=1,2,\ldots}$ [116].

In addition to the rapid convergence enabled by the preconditioned Hessian, Proposition 6 demonstrates that the computational complexity of the HPG method is maintained at $\mathcal{O}(r^2n + rn\log n)$ per iteration. This complexity is significantly more efficient than the $\mathcal{O}(n^3)$ typically encountered in common Hessian-based methods.

Computational Complexity Analysis. In this section, we demonstrate how the computational complexity of each iteration is reduced from $\mathcal{O}(n^3)$ to $\mathcal{O}(r^2n + rn\log n)$. The fundamental principle is that when $\nabla f(\boldsymbol{X}_k^e)$ is a Hankel matrix, the calculation of $\tilde{\boldsymbol{H}}_k^{-1}\nabla f(\boldsymbol{X}_k^e)$ can be efficiently performed with only $\mathcal{O}(n)$ flops. Furthermore, in cases where $\nabla f(\boldsymbol{X}_k^e)$ is not a Hankel matrix, we can employ the Neumann series expansion [147] to simplify the computation of $\tilde{\boldsymbol{H}}_k^{-1}$, as the inequality $\mathcal{I} \succ \beta\mathcal{H}\tilde{\mathcal{L}}\mathcal{H}^* \succeq 0$ holds. Consequently, to expedite the computation

of $\tilde{\boldsymbol{H}}_k^{-1}\nabla f(\boldsymbol{X}_k^e)$, we decompose $\nabla f(\boldsymbol{X}_k^e)$ based on the Hankel projection $\mathcal{HS}$. This approach ensures a significant acceleration in the calculation process.

Proposition 5. *We have*

$$\boldsymbol{X}_k^e - \tilde{\boldsymbol{H}}_k^{-1}\nabla f(\boldsymbol{X}_k^e) = \mathcal{H}\tilde{\mathcal{L}}^{-1}\mathcal{L}\mathcal{P}_\Omega \boldsymbol{s} + \beta\mathcal{H}(\tilde{\mathcal{L}}^{-1}\mathcal{L} - \mathcal{I})\mathcal{H}^*\boldsymbol{X}_k^e,$$

and this could be calculated with $\mathcal{O}(rn\log n)$ flops.

Proof. To avoid the notational cluster, we suppress the subscript k in the proof. Let $\nabla f(\boldsymbol{X}^e) = \beta(\boldsymbol{L} + \boldsymbol{X}^e)$, where $\boldsymbol{L} \coloneqq -\mathcal{H}\mathcal{L}\mathcal{P}_\Omega\boldsymbol{s} - \beta\mathcal{H}\mathcal{L}\mathcal{H}^*\boldsymbol{X}^e$ is a Hankel matrix.

$$\begin{aligned}
\tilde{\boldsymbol{H}}^{-1}\nabla f(\boldsymbol{X}^e) &= \beta^{-1}(\mathcal{I} - \beta\mathcal{H}\tilde{\mathcal{L}}\mathcal{H}^*)^{-1} \times \beta(\boldsymbol{L} + \boldsymbol{X}^e) \\
&\overset{\text{a}}{=} \sum_{k=0}^{\infty}(\beta\mathcal{H}\tilde{\mathcal{L}}\mathcal{H}^*)^k(\boldsymbol{L} + \mathcal{H}\mathcal{S}\boldsymbol{X}^e + \boldsymbol{X}^e - \mathcal{H}\mathcal{S}\boldsymbol{X}^e) \\
&\overset{\text{b}}{=} \mathcal{H}\sum_{k=0}^{\infty}(\tilde{\mathcal{L}}\beta\mathcal{W})^k\mathcal{S}(\boldsymbol{L} + \boldsymbol{X}^e) + \sum_{k=0}^{\infty}(\beta\mathcal{H}\tilde{\mathcal{L}}\mathcal{W}\mathcal{S})^k(\boldsymbol{X}^e - \mathcal{H}\mathcal{S}\boldsymbol{X}^e) \\
&\overset{\text{c}}{=} \mathcal{H}(\mathcal{I} - \tilde{\mathcal{L}}\beta\mathcal{W})^{-1}\mathcal{S}(\boldsymbol{L} + \boldsymbol{X}^e) \\
&+ (\mathcal{I} + \beta\mathcal{H}\tilde{\mathcal{L}}\mathcal{W}\mathcal{S} + (\beta\mathcal{H}\tilde{\mathcal{L}}\mathcal{W}\mathcal{S})^2 + \cdots)(\boldsymbol{X}^e - \mathcal{H}\mathcal{S}\boldsymbol{X}^e) \\
&\overset{\text{d}}{=} \mathcal{H}(\mathcal{I} - \tilde{\mathcal{L}}\beta\mathcal{H}^*\mathcal{H})^{-1}\mathcal{S}(\boldsymbol{L} + \boldsymbol{X}^e) + (\boldsymbol{X}^e - \mathcal{H}\mathcal{S}\boldsymbol{X}^e) \\
&= \mathcal{H}\tilde{\mathcal{L}}^{-1}\mathcal{S}(\boldsymbol{L} + \boldsymbol{X}^e) + (\boldsymbol{X}^e - \mathcal{H}\mathcal{S}\boldsymbol{X}^e). \qquad (6.9)
\end{aligned}$$

(a) is from $\mathcal{I} \succ \beta\mathcal{H}\tilde{\mathcal{L}}\mathcal{H}^* \succeq 0$ and the Neumann series [147];

(b) results from that $(\boldsymbol{L} + \mathcal{H}\mathcal{S}\boldsymbol{X}^e)$ is Hankel, and $(\mathcal{H}\mathcal{S})^k = \mathcal{H}\mathcal{S}$ for $k \geq 1$;

(c) is becaues of Neumann series as $\sum_{k=0}^{\infty}(\tilde{\mathcal{L}}\beta\mathcal{W})^k = (\mathcal{I} - \tilde{\mathcal{L}}\beta\mathcal{W})^{-1}$.

(d) is noticed by the fact that $\beta\mathcal{H}\tilde{\mathcal{L}}\mathcal{W}\mathcal{S}(\boldsymbol{X}^e - \mathcal{H}\mathcal{S}\boldsymbol{X}^e) = \beta\mathcal{H}\tilde{\mathcal{L}}\mathcal{W}\mathcal{S}\boldsymbol{X}^e - \beta\mathcal{H}\tilde{\mathcal{L}}\mathcal{W}\mathcal{S}\boldsymbol{X}^e = 0$.

Then we have

$$\begin{aligned}\boldsymbol{X}_k^e - \tilde{\boldsymbol{H}}_k^{-1}\nabla f(\boldsymbol{X}_k^e) &= -\beta\mathcal{H}\mathcal{H}^*\boldsymbol{X}_k^e - \mathcal{H}\tilde{\mathcal{L}}^{-1}\mathcal{S}\boldsymbol{L}_k \\ &= \mathcal{H}\tilde{\mathcal{L}}^{-1}\mathcal{L}\mathcal{P}_\Omega\boldsymbol{s} + \beta\mathcal{H}(\tilde{\mathcal{L}}^{-1}\mathcal{L} - \mathcal{I})\mathcal{H}^*\boldsymbol{X}_k^e. \end{aligned} \tag{6.10}$$

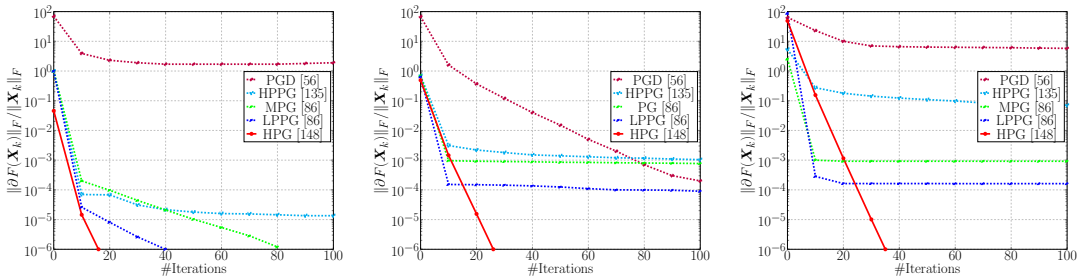
Therefore, the computational burden of $\boldsymbol{X}_k^e - \tilde{\boldsymbol{H}}_k^{-1}\nabla f(\boldsymbol{X}_k^e)$ is mainly from $\mathcal{H}^*\boldsymbol{X}_k^e$. Since $\boldsymbol{X}_k^e$ is a low-rank matrix, $\mathcal{H}^*\boldsymbol{X}_k^e$ can be computed in $\mathcal{O}(rn\log n)$ flops via FFT [51]. □

Proposition 6. *The computational complexity of the HPG method is $\mathcal{O}(r^2n + rn\log n)$ per iteration.*

Proof. As established in Proposition 5, the forward step in HPG has a complexity of $\mathcal{O}(rn\log n)$. Subsequent to this, the proximal mapping in HPG, which either operates as a truncated SVD operator on a Hankel matrix (refer to the Hessian proximal step in Algorithm 9) or as a linear combination of a Hankel matrix and a low-rank matrix (refer to the standard proximal step), can be expedited. This acceleration is achieved through the Lanczos method, as detailed in [51], which requires $\mathcal{O}(r^2n + rn\log n)$ flops. Moreover, the update condition outlined in (6.8) involves an inner product calculation, also computable within $\mathcal{O}(r^2n + rn\log n)$ flops. Consequently, the overall per-iteration computational complexity of the HPG method is $\mathcal{O}(r^2n + rn\log n)$. □

6.4 Experiments

In this section, we demonstrate the performance of the HPG as compared to the modified PG (MPG) [86], the LPPG [86], the PGD [56] and the HPPG [135]. The stopping criteria are set as $K = 1000$ and $\epsilon = 10^{-6}$, where K is the maximum number of iterations and ϵ is the numerical-error threshold. In addition, the frequency f_k of SSS is randomly generated according to the uniform distribution on $[0, 1)$, and the phase of complex coefficients b_k is uniformly sampled on $[0, 2\pi)$ while the amplitudes are selected to be $1 + 10^{0.5c_k}$ where c_k is uniformly distributed on $[0, 1]$; cf. [56]. For a given sample size $m = S_p n$, it is assumed that Ω is uniformly sampled. In order to study the robustness in practice, the samples contaminated by the zero-mean Gaussian



(a) $\beta = 0.01$ (b) $\beta = 1$ (c) $\beta = 100$

Figure 6.1: Convergence speed for different β: $\boldsymbol{x} \in \mathbb{C}^{2001}$, the resulting $\boldsymbol{X}^e \in \mathbb{C}^{1001\times 1001}$, $S_p = 0.5$, $r = 20$, and SNR = 10dB.

noise are considered. The computational efficiency (time) and the normalized mean squared error (which is defined as $\|\boldsymbol{x}^\dagger - \boldsymbol{x}\|_2/\|\boldsymbol{x}\|_2$ where $\boldsymbol{x}^\dagger$ denotes the estimate returned by HPG; NMSE) are considered to show the advantages of our HPG method. The results are averaged on 50 independent Monte–Carlo trials. The Hankel enforcement parameter β is optimized for each algorithm and noise level to pursue the best reconstruction.

6.4.1 Convergence Analysis of the HPG Method

In this subsection, we demonstrate the notable efficiency of the HPG method, particularly its ability to achieve rapid convergence with fewer iterations. This aspect is critical, considering the usual trade-off between reconstruction accuracy and computational efficiency in SCS algorithms. Our analysis reveals that the choice of the Hankel enforcement parameter β significantly influences the convergence speed and reconstruction accuracy. As illustrated in Table 6.1, a larger β value typically results in superior reconstruction performance in practical scenarios but at the cost of slower convergence. This slower convergence often leads to an early termination at the maximum iteration limit of $k = 1000$. Conversely, while the PGD method shows commendable performance for $\beta = 1$, the MPG, LPPG, and HPPG methods tend to yield better outcomes with a smaller β value of 0.01. The process of fine-tuning β for varying datasets and noise levels can be laborious and inefficient. In stark contrast, the HPG method exhibits remarkable adaptability, efficiently converging to critical points across a broad spectrum of β values without entailing substantial additional computational costs. This attribute of HPG is notably beneficial

in scenarios with variable noise levels and data characteristics. Figure 6.1 further supports the fast convergence speed of HPG. Notably, the convergence rate is directly measured using the subgradient, which provides a more accurate and reliable metric for assessing the algorithm's performance in reaching critical points.

Table 6.1: NMSEs and # Iter for different β: $\boldsymbol{x} \in \mathbb{C}^{2001}$, the resulting $\boldsymbol{X}^e \in \mathbb{C}^{1001\times 1001}$, $S_p = 0.5$, $r = 20$, and SNR = 10dB.

β	0.01		1		100	
Criterion	NMSE	Iter	NMSE	Iter	NMSE	Iter
PGD [56]	4.6e-1	1000	8.5e-2	176	1.2e-1	1000
MPG [86]	9.6e-2	78	2.1e-1	1000	5.1e-1	1000
LPPG [86]	9.6e-2	42	7.0e-2	1000	1.2e-1	1000
HPPG [135]	9.6e-2	73	1.1e-1	1000	1.3e-1	1000
HPG [148]	8.0e-2	16	8.0e-2	28	7.9e-2	33

6.4.2 Overall Performance of HPG in Terms of Time Efficiency and NMSE

This section highlights the comprehensive performance advantages of the HPG method, focusing on time efficiency and NMSE metrics. Based on the earlier discussions, we adopt a default setting of $\beta = 100$ for the HPG algorithm, which has been shown to deliver superior NMSEs consistently. The choice of a larger, fixed value for β simplifies the implementation of HPG, bypassing the need for extensive tuning processes typically associated with other methods. This default setting is crucial for demonstrating the robustness and adaptability of HPG across different noise levels and problem scenarios.

Table 6.2 presents a comparative analysis of both NMSEs and wall-clock times for various algorithms, considering both fixed and optimized β values. Notably, the time-intensive process of tuning β for other algorithms is excluded from this comparison, emphasizing the inherent efficiency of HPG. The results from Table 6.2 reveal that the HPG algorithm consistently surpasses other methods in achieving lower NMSEs while maintaining superior computational

efficiency. This advantage is accentuated by the reduced number of iterations and the lower computational complexity per iteration inherent in the HPG method. The findings underscore the practicality and effectiveness of HPG, particularly in applications where time efficiency and reconstruction accuracy are paramount.

Table 6.2: NMSEs and time(s) for different SNR with fixed β: $\boldsymbol{x} \in \mathbb{C}^{2001}$, the resulting $\boldsymbol{X}^e \in \mathbb{C}^{1001\times 1001}$, $S_p = 0.5$, and $r = 20$.

SNR	0dB		10dB		20dB	
Criterion	NMSE	Time	NMSE	Time	NMSE	Time
PGD [56]	2.6e-1	3.8	9.1e-2	3.1	2.6e-2	3.2
MPG [86]	3.1e-1	4.4e-1	1.8e-1	7.9e-1	5.1e-2	9.1e-1
LPPG [86]	3.1e-1	8.4	1.0e-1	8.1	3.0e-2	8.4
HPPG [135]	3.1e-1	4.4e-1	1.1e-1	3.5e-1	3.5e-2	1.0
HPG [148]	2.5e-1	1.5	8.9e-2	1.7	2.5e-2	1.8

6.4.3 Robustness of Model Order in HPG Method

In this subsection, we explore the robustness of the HPG method regarding model order estimation, building upon observations made in Chapter 4. As noted previously, overestimating the model order results in acceptable reconstruction errors; however, it significantly increases the number of iterations required for convergence. The experimental setup parallels that in Chapter 4, with a focus on three-dimensional spectrally sparse signals. The true model order is set at $r = 5$ with a sampling rate of 0.5. We systematically vary the model order, starting from $r = 2$ and incrementally increasing it by 1 until reaching a maximum of $r = 8$. This approach allows us to evaluate the performance of the HPG method under various model order estimations. Three distinct levels of additive noise are considered, corresponding to optimized Hankel enforcement parameters. The termination of the tested algorithms is contingent upon two conditions: a relative change in the Hankel matrix below 10^{-6} ($\|\boldsymbol{X}^e_{k+1} - \boldsymbol{X}^e_k\|_F / \|\boldsymbol{X}^e_k\|_F$) or reaching the maximum iteration count of 1000.

The results, as summarized in Table 6.3, demonstrate the robustness of the HPG method in scenarios involving both high noise levels (large Hankel enforcement parameter) and overesti-

Table 6.3: NMSE and Iter for undamped Signals: $\boldsymbol{X} \in \mathbb{C}^{11\times11\times11}$, $\boldsymbol{X}^e \in \mathbb{C}^{216\times216}$ and $S_p = 0.5$.

Test rank		2	3	4	5	6	7	8
	SNR = ∞							
FIHT [57]	NMSE	0.66	0.47	0.29	2.6e-7	1.0e-2	1.4e-2	1.9e-2
	Iter	12	13	13	14	1000	1000	1000
PGD [56]	NMSE	0.66	0.47	0.29	6.6e-6	2.3e-5	8.0e-4	1.0e-3
	Iter	278	96	154	53	872	1000	1000
Modified PG [86]	NMSE	0.50	0.34	0.22	1.0e-6	4.5e-4	1.3e-3	1.5e-3
	Iter	25	26	28	29	53	53	52
LPPG [86]	NMSE	0.50	0.34	0.22	1.0e-6	1.0e-6	1.0e-6	1.0e-6
	Iter	17	18	19	17	18	18	18
HPPG [135]	NMSE	0.45	0.34	0.21	1.2e-7	2.1e-2	2.6e-2	3.6e-2
	Iter	9	21	13	14	95	426	1000
HPG [148]	NMSE	0.46	0.34	0.22	1.3e-6	8.9e-4	2.9e-3	3.3e-3
	Iter	21	21	24	24	28	45	53
	SNR = 20							
FIHT [57]	NMSE	0.67	0.49	0.31	2.2e-2	2.4e-2	2.9e-2	3.3e-2
	Iter	102	102	125	124	281	292	732
PGD [56]	NMSE	0.67	0.49	0.37	2.2e-2	2.2e-2	2.3e-2	2.8e-2
	Iter	215	309	429	351	1000	1000	1000
Modified PG [86]	NMSE	0.63	0.46	0.29	1.9e-2	2.1e-2	2.5e-2	2.7e-2
	Iter	366	354	392	386	566	443	1000
LPPG [86]	NMSE	0.63	0.46	0.29	1.9e-2	2.1e-2	2.5e-2	2.8e-2
	Iter	119	136	152	185	216	233	332
HPPG [135]	NMSE	0.66	0.47	0.30	1.5e-2	2.1e-2	2.3e-2	2.3e-2
	Iter	1000	288	294	118	199	276	332
HPG [148]	NMSE	0.66	0.47	0.30	2.2e-2	2.6e-2	3.3e-2	3.7e-2
	Iter	20	21	23	24	36	37	79
	SNR = 0							
FIHT [57]	NMSE	0.67	0.51	0.36	0.22	0.26	0.30	0.33
	Iter	108	117	123	119	724	1000	1000
PGD [56]	NMSE	0.67	0.51	0.36	0.22	0.40	0.30	0.33
	Iter	289	312	444	356	1000	1000	1000
Modified PG [86]	NMSE	0.65	0.49	0.34	0.18	0.21	0.25	0.27
	Iter	377	378	382	380	1000	1000	1000
LPPG [86]	NMSE	0.65	0.49	0.34	0.16	0.21	0.29	0.29
	Iter	172	175	189	192	556	566	581
HPPG [135]	NMSE	0.66	0.49	0.33	0.14	0.18	0.23	0.28
	Iter	168	178	284	462	1000	1000	1000
HPG [148]	NMSE	0.67	0.49	0.36	0.23	0.25	0.30	0.34
	Iter	21	22	23	24	55	77	79

mated model orders. This finding suggests that when reconstruction accuracy is not the primary concern, setting a larger β value and a higher-than-actual model order can be a viable strategy. This approach can offer computational advantages, particularly in practical scenarios where the

exact model order is unknown or difficult to estimate accurately.

6.5 Conclusions

In this chapter, we introduce the HPG method, a novel approach tailored for SCS with Hessian information. The primary innovation of the HPG method lies in its integration of Hessian information to accelerate the convergence towards optimal solutions maintaining acceptable computational complexity. Specifically, the method:

- Utilizes the Hankel and low-rank structures to minimize both the number of iterations needed for convergence and the computational complexity per iteration.
- Demonstrates a marked improvement in recovery accuracy and computational efficiency compared to current state-of-the-art approaches in scenarios involving both high noise levels (large Hankel enforcement parameter) and overestimated model orders.

Through empirical studies, the HPG method has been shown to outperform existing methods in SCS. The advantages are particularly notable in terms of recovery accuracy and computational speed, making the HPG method a preferred choice for practical applications in this domain.

Part III

Summary and Conclusions

Chapter 7

Conclusion

In this chapter, we will summarize the thesis achievements and discuss the future work.

7.1 Summary of Thesis Achievements

This thesis has delved into various proximal-type approaches for the recovery of spectrally sparse signals via unweighted multi-block Hankel matrix form. Key achievements are outlined below:

7.1.1 Fast Computation of Multi-block Hankel Matrix

Chapter 2 reviews the groundwork with background of super-resolution SCS including two typical convex ANM and EMaC algorithms. In Chapter 3, we initiate our formal study by introducing fast algorithms for computing the structured Hankel matrix. These algorithms, designed for multi-level and large-scale data, capitalize on the compact Hankel structure and the FFT. They significantly reduce computational complexity, making them highly practical. Our exploration includes algorithms for Hankel operator computation, matrix-vector multiplication, truncated singular value decomposition, and Penrose Pseudoinverse of Hankel operator. Additionally, we have developed a Julia package tailored specifically for SCS. Numerical results demonstrate the

effectiveness of these algorithms.

7.1.2 Fast Proximal-type Algorithms for Spectral Compressed Sensing

Chapter 4 introduces our novel algorithm, the low-rank projected proximal gradient (LPPG), designed for recovering spectrally sparse signals from unweighted multi-block Hankel matrices. This algorithm is built upon a non-convex non-smooth optimization formulation and leverages modified proximal gradient methods and low-rank projection to address convergence speed issues. It achieves faster convergence and empirically superior reconstruction accuracy compared to existing methods. Extensive numerical results support these claims, highlighting LPPG's efficiency in terms of accuracy and computation time.

In Chapter 5, we retain the beneficial aspects of our formulation while enhancing convergence speed. Despite LPPG's superior recovery performance, its computational complexity limits its practicality for large-scale data with high model order. To address this, we utilize the Hankel structure to increase the step size inversely proportional to the sampling ratio and bound the Lipschitz constant, regardless of the size of the Hankel enforcement parameter β. This modification reduces computational complexity from $\mathcal{O}(r^4n + r^3n\log n)$ to $\mathcal{O}(r^2n + rn\log n)$, as demonstrated by our numerical results.

Chapter 6 explores the use of second-order information to further accelerate convergence. The Hessian proximal gradient (HPG) method, developed here, employs a preconditioned Hessian matrix to significantly improve convergence, especially in challenging scenarios with high noise levels and overestimated model order. By leveraging the Hankel structure, we reduce the computational complexity of the Hessian matrix and ensure convergence without solving complex subproblems. Numerical evidence suggests that HPG surpasses LPPG and HPPG in computational efficiency, particularly when reconstruction accuracy is less critical.

7.1.3 Comparison of the Proposed Algorithms

In this thesis, we proposed three algorithms for the recovery of spectrally sparse signals from unweighted multi-block Hankel matrices. The structured enhanced matrix completion approaches, which excel in handling large signal sizes while achieving super-resolution, are summarized as follows:

- Structured Enhanced Matrix Completion
  - Nonconvex
    - Weighted Norm
      - Gradient Descent (PGD) [56]
      - Alternating Projection (Cadzow) [57]
    - Unweighted Norm
      - Low-Rank Projected Proximal Gradient [86]
      - Hankel Projected Proximal Gradient [135]
      - Hessian Proximal Gradient [148]
  - Convex
    - Semi-Definite Program [35]
    - Soft Value Thresholding [2]

(7.1)

The unweighted norm approach potentially offers better reconstruction accuracy due to unbiased error measurement. Our three modified proximal gradient algorithms are all based on this principle and successfully address the issues of size-dependent slow convergence. Each algorithm employs different optimization acceleration techniques, making it challenging to definitively identify the best among them. The advantages and disadvantages of these three algorithms are summarized in Table 7.1.

Table 7.1: Comparison of the Proposed Algorithms.

Method Name	**Features**	**Computational Complexity per Iteration**	**Advantages**	**Disadvantages**
LPPG	Low-rank projection	$\mathcal{O}(r^4 n + r^3 n \log n)$	Excellent reconstruction accuracy and phase-transition performance	High computational cost for large model orders and high noise levels
HPPG	Hankel structure projection with improved step-size	$\mathcal{O}(r^2 n + rn \log n)$	Rapid convergence and robust performance	Slower convergence under high noise and with over-estimated model order
HPG	Hessian matrix utilization for second-order method	$\mathcal{O}(r^2 n + rn \log n)$	Fast convergence in challenging scenarios	Slightly lower reconstruction accuracy compared to LPPG and HPPG

7.2 Future Work Directions

This thesis has made significant contributions to signal recovery guarantees and fast convergence rate analysis. However, there remain avenues for further research and improvement.

7.2.1 Signal Recovery Guarantee

The empirical results in this thesis demonstrate the phase transition of exact recovery and the high reconstruction accuracy of the proposed methods. While convergence to critical points is theoretically proven, a comprehensive theoretical analysis of phase transition recovery and reconstruction accuracy in the nonconvex setting is still needed. Future work will aim to rigorously establish these aspects, particularly focusing on array form and incoherence conditions, which are critical for understanding and improving the performance of these recovery methods.

7.2.2 Enhancing Fast Convergence

The optimization problem in this thesis is formulated as a nonconvex and nonsmooth challenge, for which proximal-type methods have been developed to ensure convergence. However, there are other acceleration techniques, such as quasi-Newton methods and Riemannian manifold optimization, that warrant exploration. Drawing from the successful implementation of Hankel structure in the HPG approach, it is anticipated that these advanced methods could be further simplified and enhanced by leveraging the Hankel structure. Additionally, a detailed analysis of the fast convergence rates of the proposed algorithms is a crucial area for future investigation, potentially leading to more efficient and robust signal recovery solutions.

Bibliography

[1] Y. Chi and M. F. Da Costa, "Harnessing sparsity over the continuum: Atomic norm minimization for superresolution," *IEEE Signal Processing Magazine*, vol. 37, no. 2, pp. 39–57, 2020.

[2] Y. Chen and Y. Chi, "Spectral compressed sensing via structured matrix completion," in *International Conference on Machine Learning*. PMLR, 2013, pp. 414–422.

[3] L. Lu, W. Xu, and S. Qiao, "A fast SVD for multilevel block Hankel matrices with minimal memory storage," *Numerical Algorithms*, vol. 69, no. 4, pp. 875–891, 2015.

[4] G. Golub and W. Kahan, "Calculating the singular values and pseudo-inverse of a matrix," *Journal of the Society for Industrial and Applied Mathematics, Series B: Numerical Analysis*, vol. 2, no. 2, pp. 205–224, 1965.

[5] J. Baglama and L. Reichel, "Augmented implicitly restarted lanczos bidiagonalization methods," *SIAM Journal on Scientific Computing*, vol. 27, no. 1, pp. 19–42, 2005.

[6] N. Parikh, S. Boyd, *et al.*, "Proximal algorithms," *Foundations and trends® in Optimization*, vol. 1, no. 3, pp. 127–239, 2014.

[7] J. D. Lee, Y. Sun, and M. Saunders, "Proximal Newton-type methods for convex optimization," *Advances in Neural Information Processing Systems*, vol. 25, 2012.

[8] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[9] M. F. Duarte and R. G. Baraniuk, "Spectral compressive sensing," *Applied and Computational Harmonic Analysis*, vol. 35, no. 1, pp. 111–129, 2013.

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

[11] L. C. Potter, E. Ertin, J. T. Parker, and M. Cetin, "Sparsity and compressed sensing in radar imaging," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1006–1020, 2010.

[12] L. Schermelleh, R. Heintzmann, and H. Leonhardt, "A guide to super-resolution fluorescence microscopy," *Journal of Cell Biology*, vol. 190, no. 2, pp. 165–175, 2010.

[13] Y. Chi, Y. Xie, and R. Calderbank, "Compressive demodulation of mutually interfering signals," *arXiv preprint arXiv:1303.3904*, 2013.

[14] J. Gillard and K. Usevich, "Hankel low-rank approximation and completion in time series analysis and forecasting: a brief review," *arXiv preprint arXiv:2206.05103*, 2022.

[15] X. Qu, M. Mayzel, J.-F. Cai, Z. Chen, and V. Orekhov, "Accelerated nmr spectroscopy with low-rank reconstruction," *Angewandte Chemie International Edition*, vol. 54, no. 3, pp. 852–854, 2015.

[16] I. F. Gorodnitsky and B. D. Rao, "Sparse signal reconstruction from limited data using focuss: A re-weighted minimum norm algorithm," *IEEE Transactions on Signal Processing*, vol. 45, no. 3, pp. 600–616, 1997.

[17] D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[18] T. Blumensath and M. E. Davies, "Iterative hard thresholding for compressed sensing," *Applied and Computational Harmonic Analysis*, vol. 27, no. 3, pp. 265–274, 2009.

[19] W. Dai and O. Milenkovic, "Subspace pursuit for compressive sensing signal reconstruction," *IEEE Transactions on Information Theory*, vol. 55, no. 5, pp. 2230–2249, 2009.

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

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

[22] M. A. Herman and T. Strohmer, "General deviants: An analysis of perturbations in compressed sensing," *IEEE Journal of Selected topics in Signal Processing*, vol. 4, no. 2, pp. 342–349, 2010.

[23] D. Nion and N. D. Sidiropoulos, "Tensor algebra and multidimensional harmonic retrieval in signal processing for mimo radar," *IEEE Transactions on Signal Processing*, vol. 58, no. 11, pp. 5693–5705, 2010.

[24] L. Borcea, G. Papanicolaou, C. Tsogka, and J. Berryman, "Imaging and time reversal in random media," *Inverse Problems*, vol. 18, no. 5, p. 1247, 2002.

[25] A. M. Sayeed and B. Aazhang, "Joint multipath-doppler diversity in mobile wireless communications," *IEEE Transactions on Communications*, vol. 47, no. 1, pp. 123–132, 1999.

[26] K. Gedalyahu, R. Tur, and Y. C. Eldar, "Multichannel sampling of pulse streams at the rate of innovation," *IEEE Transactions on Signal Processing*, vol. 59, no. 4, pp. 1491–1504, 2011.

[27] P. Stoica, R. L. Moses, *et al.*, *Spectral analysis of signals*. Pearson Prentice Hall Upper Saddle River, NJ, 2005, vol. 452.

[28] H. J. Nussbaumer and H. J. Nussbaumer, *The fast Fourier transform*. Springer, 1982.

[29] R. Schmidt, "Multiple emitter location and signal parameter estimation," *IIEEE Transactions on Antennas and Propagation*, vol. 34, no. 3, pp. 276–280, 1986.

[30] G. R. de Prony, "Essai experimental et analytique: Sur les lois de la dilatabilite des fluides elastique et sur celles de la force expansive de la vapeur de l'eau et de la vapeur de l'alkool, a differentes temperatures," *J. de l'Ecole Polytechnique*, 1795.

[31] Y. Hua, "Estimating two-dimensional frequencies by matrix enhancement and matrix pencil," *IEEE Transactions on Signal Processing*, vol. 40, no. 9, pp. 2267–2280, 1992.

[32] P. L. Dragotti, M. Vetterli, and T. Blu, "Sampling moments and reconstructing signals of finite rate of innovation: Shannon meets strang–fix," *IEEE Transactions on Signal Processing*, vol. 55, no. 5, pp. 1741–1757, 2007.

[33] E. J. Candès and C. Fernandez-Granda, "Towards a mathematical theory of super-resolution," *Communications on pure and applied Mathematics*, vol. 67, no. 6, pp. 906–956, 2014.

[34] ——, "Super-resolution from noisy data," *Journal of Fourier Analysis and Applications*, vol. 19, pp. 1229–1254, 2013.

[35] G. Tang, B. N. Bhaskar, P. Shah, and B. Recht, "Compressed sensing off the grid," *IEEE Transactions on Information Theory*, vol. 59, no. 11, pp. 7465–7490, 2013.

[36] V. Chandrasekaran, B. Recht, P. A. Parrilo, and A. S. Willsky, "The convex geometry of linear inverse problems," *Foundations of Computational mathematics*, vol. 12, pp. 805–849, 2012.

[37] B. N. Bhaskar, G. Tang, and B. Recht, "Atomic norm denoising with applications to line spectral estimation," *IEEE Transactions on Signal Processing*, vol. 61, no. 23, pp. 5987–5999, 2013.

[38] J.-M. Azais, Y. De Castro, and F. Gamboa, "Spike detection from inaccurate samplings," *Applied and Computational Harmonic Analysis*, vol. 38, no. 2, pp. 177–195, 2015.

[39] Z. Yang, L. Xie, and P. Stoica, "Vandermonde decomposition of multilevel toeplitz matrices with application to multidimensional super-resolution," *IEEE Transactions on Information Theory*, vol. 62, no. 6, pp. 3685–3701, 2016.

[40] Z. Yang and L. Xie, "On gridless sparse methods for line spectral estimation from complete and incomplete data," *IEEE Transactions on Signal Processing*, vol. 63, no. 12, pp. 3139–3153, 2015.

[41] X. Wu, W.-P. Zhu, and J. Yan, "A toeplitz covariance matrix reconstruction approach for direction-of-arrival estimation," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 9, pp. 8223–8237, 2017.

[42] C. Zhou, Y. Gu, X. Fan, Z. Shi, G. Mao, and Y. D. Zhang, "Direction-of-arrival estimation for coprime array via virtual array interpolation," *IEEE Transactions on Signal Processing*, vol. 66, no. 22, pp. 5956–5971, 2018.

[43] X. Wu, Z. Yang, P. Stoica, and Z. Xu, "Maximum likelihood line spectral estimation in the signal domain: A rank-constrained structured matrix recovery approach," *IEEE Transactions on Signal Processing*, vol. 70, pp. 4156–4169, 2022.

[44] X. Wu, Z. Yang, J.-F. Cai, and Z. Xu, "Spectral super-resolution on the unit circle via gradient descent," in *International Conference on Acoustics, Speech, and Signal Processing*. IEEE, 2023, pp. 1–5.

[45] W. Xu, J. Yi, S. Dasgupta, J.-F. Cai, M. Jacob, and M. Cho, "Seperation-free super-resolution from compressed measurements is possible: an orthonormal atomic norm minimization approach," in *2018 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2018, pp. 76–80.

[46] Y. Chi, "Convex relaxations of spectral sparsity for robust super-resolution and line spectrum estimation," in *Wavelets and Sparsity XVII*, vol. 10394. SPIE, 2017, pp. 314–321.

[47] W. Liao and A. Fannjiang, "Music for single-snapshot spectral estimation: Stability and super-resolution," *Applied and Computational Harmonic Analysis*, vol. 40, no. 1, pp. 33–67, 2016.

[48] N. Gillis and F. Glineur, "Low-rank matrix approximation with weights or missing data is np-hard," *SIAM Journal on Matrix Analysis and Applications*, vol. 32, no. 4, pp. 1149–1165, 2011.

[49] G. Ottaviani, P.-J. Spaenlehauer, and B. Sturmfels, "Exact solutions in structured low-rank approximation," *SIAM Journal on Matrix Analysis and Applications*, vol. 35, no. 4, pp. 1521–1542, 2014.

[50] E. Candes and B. Recht, "Exact matrix completion via convex optimization," *Communications of the ACM*, vol. 55, no. 6, pp. 111–119, 2012.

[51] G. H. Golub and C. F. Van Loan, *Matrix computations*. JHU press, 2013.

[52] S. Burer and R. D. Monteiro, "Local minima and convergence in low-rank semidefinite programming," *Mathematical programming*, vol. 103, no. 3, pp. 427–444, 2005.

[53] X. Zhang, Y. Liu, and W. Cui, "Spectrally sparse signal recovery via Hankel matrix completion with prior information," *IEEE Transactions on Signal Processing*, vol. 69, pp. 2174–2187, 2021.

[54] J. A. Cadzow, "Signal enhancement-a composite property mapping algorithm," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 36, no. 1, pp. 49–62, 1988.

[55] J. Ying, H. Lu, Q. Wei, J.-F. Cai, D. Guo, J. Wu, Z. Chen, and X. Qu, "Hankel matrix nuclear norm regularized tensor completion for n-dimensional exponential signals," *IEEE Transactions on Signal Processing*, vol. 65, no. 14, pp. 3702–3717, 2017.

[56] J.-F. Cai, T. Wang, and K. Wei, "Spectral compressed sensing via projected gradient descent," *SIAM Journal on Optimization*, vol. 28, no. 3, pp. 2625–2653, 2018.

[57] ——, "Fast and provable algorithms for spectrally sparse signal reconstruction via low-rank hankel matrix completion," *Applied and Computational Harmonic Analysis*, vol. 46, no. 1, pp. 94–121, 2019.

[58] H. Wang, J.-F. Cai, T. Wang, and K. Wei, "Fast cadzow's algorithm and a gradient variant," *Journal of Scientific Computing*, vol. 88, no. 2, pp. 1–21, 2021.

[59] É. Schost and P.-J. Spaenlehauer, "A quadratically convergent algorithm for structured low-rank approximation," *Foundations of Computational Mathematics*, vol. 16, no. 2, pp. 457–492, 2016.

[60] A. S. Lewis and J. Malick, "Alternating projections on manifolds," *Mathematics of Operations Research*, vol. 33, no. 1, pp. 216–234, 2008.

[61] F. Andersson, M. Carlsson, J.-Y. Tourneret, and H. Wendt, "A new frequency estimation method for equally and unequally spaced data," *IEEE Transactions on Signal Processing*, vol. 62, no. 21, pp. 5761–5774, 2014.

[62] K. Wei, J.-F. Cai, T. F. Chan, and S. Leung, "Guarantees of riemannian optimization for low rank matrix completion," *arXiv preprint arXiv:1603.06610*, 2016.

[63] J. W. Gillard and A. A. Zhigljavsky, "Stochastic algorithms for solving structured low-rank matrix approximation problems," *Communications in Nonlinear Science and Numerical Simulation*, vol. 21, no. 1-3, pp. 70–88, 2015.

[64] N. Zvonarev and N. Golyandina, "Iterative algorithms for weighted and unweighted finite-rank time-series approximations," *arXiv preprint arXiv:1507.02751*, 2015.

[65] J. Fang, F. Wang, Y. Shen, H. Li, and R. S. Blum, "Super-resolution compressed sensing for line spectral estimation: An iterative reweighted approach," *IEEE Transactions on Signal Processing*, vol. 64, no. 18, pp. 4649–4662, 2016.

[66] R. Kumaresan, D. W. Tufts, and L. L. Scharf, "A prony method for noisy data: Choosing the signal components and selecting the order in exponential signal models," *Proceedings of the IEEE*, vol. 72, no. 2, pp. 230–233, 1984.

[67] R. Roy and T. Kailath, "Esprit-estimation of signal parameters via rotational invariance techniques," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 7, pp. 984–995, 1989.

[68] M. A. Herman and T. Strohmer, "High-resolution radar via compressed sensing," *IEEE Transactions on Signal Processing*, vol. 57, no. 6, pp. 2275–2284, 2009.

[69] R. Sanyal, F. Sottile, and B. Sturmfels, "Orbitopes," *Mathematika*, vol. 57, no. 2, pp. 275–314, 2011.

[70] J. B. Lasserre, *Moments, positive polynomials and their applications*. World Scientific, 2009, vol. 1.

[71] B. Dumitrescu, *Positive trigonometric polynomials and signal processing applications*. Springer, 2007, vol. 103.

[72] T. T. Georgiou, "The carathéodory–fejér–pisarenko decomposition and its multivariable counterpart," *IEEE Transactions on automatic control*, vol. 52, no. 2, pp. 212–228, 2007.

[73] M. Grant and S. Boyd, "Cvx: Matlab software for disciplined convex programming, version 2.1," 2014.

[74] D. L. Boley, F. T. Luk, and D. Vandevoorde, "A fast method to diagonalize a hankel matrix," *Linear algebra and its applications*, vol. 284, no. 1-3, pp. 41–52, 1998.

[75] E. J. Candes, "The restricted isometry property and its implications for compressed sensing," *Comptes rendus. Mathematique*, vol. 346, no. 9-10, pp. 589–592, 2008.

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

[77] G. Tang, B. N. Bhaskar, and B. Recht, "Near minimax line spectral estimation," *IEEE Transactions on Information Theory*, vol. 61, no. 1, pp. 499–512, 2014.

[78] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society Series B: Statistical Methodology*, vol. 58, no. 1, pp. 267–288, 1996.

[79] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, *et al.*, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine learning*, vol. 3, no. 1, pp. 1–122, 2011.

[80] Y. Li, J. Razavilar, and K. R. Liu, "A high-resolution technique for multidimensional nmr spectroscopy," *IEEE Transactions on Biomedical Engineering*, vol. 45, no. 1, pp. 78–86, 1998.

[81] J. Gao, M. D. Sacchi, and X. Chen, "A fast reduced-rank interpolation method for prestack seismic volumes that depend on four spatial dimensions," *Geophysics*, vol. 78, no. 1, pp. V21–V30, 2013.

[82] D. Gross, "Recovering low-rank matrices from few coefficients in any basis," *IEEE Transactions on Information Theory*, vol. 57, no. 3, pp. 1548–1566, 2011.

[83] 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.

[84] J.-F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM Journal on optimization*, vol. 20, no. 4, pp. 1956–1982, 2010.

[85] Z. Yang and X. Wu, "New low rank optimization model and convex approach for robust spectral compressed sensing," *arXiv preprint arXiv:2101.06433*, 2021.

[86] X. Yao and W. Dai, "A projected proximal gradient method for efficient recovery of spectrally sparse signals," in *2023 31st European Signal Processing Conference (EUSIPCO)*. IEEE, 2023, pp. 1738–1742.

[87] N. Golyandina, V. Nekrutkin, and A. A. Zhigljavsky, *Analysis of time series structure: SSA and related techniques*. CRC press, 2001.

[88] M. A. Davenport and J. Romberg, "An overview of low-rank matrix recovery from incomplete observations," *IIEEE Journal of Selected Topics in Signal Processing*, vol. 10, no. 4, pp. 608–622, 2016.

[89] R.-P. Wen, S.-Z. Li, and F. Zhou, "Toeplitz matrix completion via smoothing augmented lagrange multiplier algorithm," *Applied Mathematics and Computation*, vol. 355, pp. 299–310, 2019.

[90] M. Fazel, T. K. Pong, D. Sun, and P. Tseng, "Hankel matrix rank minimization with applications to system identification and realization," *SIAM Journal on Matrix Analysis and Applications*, vol. 34, no. 3, pp. 946–977, 2013.

[91] V. Cevher, S. Becker, and M. Schmidt, "Convex optimization for big data: Scalable, randomized, and parallel algorithms for big data analytics," *IEEE Signal Processing Magazine*, vol. 31, no. 5, pp. 32–43, 2014.

[92] E. J. Candes, Y. C. Eldar, T. Strohmer, and V. Voroninski, "Phase retrieval via matrix completion," *SIAM review*, vol. 57, no. 2, pp. 225–251, 2015.

[93] N. Boumal, V. Voroninski, and A. Bandeira, "The non-convex burer-monteiro approach works on smooth semidefinite programs," *Advances in Neural Information Processing Systems*, vol. 29, 2016.

[94] P. Jain, P. Kar, *et al.*, "Non-convex optimization for machine learning," *Foundations and Trends® in Machine Learning*, vol. 10, no. 3-4, pp. 142–363, 2017.

[95] S. P. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[96] H. Bauschke and P. Combettes, "Convex analysis and monotone operator theory in hilbert spaces, 2011," *CMS books in mathematics). DOI*, vol. 10, pp. 978–1, 2010.

[97] P. L. Combettes and J.-C. Pesquet, "Proximal splitting methods in signal processing," *Fixed-point algorithms for inverse problems in science and engineering*, pp. 185–212, 2011.

[98] F. Facchinei and J.-S. Pang, *Finite-dimensional variational inequalities and complementarity problems*. Springer, 2003.

[99] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM Journal on Imaging Sciences*, vol. 2, no. 1, pp. 183–202, 2009.

[100] Y. E. Nesterov, "A method of solving a convex programming problem with convergence rate o\bigl(k^2\bigr)," in *Doklady Akademii Nauk*, vol. 269, no. 3. Russian Academy of Sciences, 1983, pp. 543–547.

[101] J. F. Bonnans, J. C. Gilbert, C. Lemaréchal, and C. A. Sagastizábal, "A family of variable metric proximal methods," *Mathematical Programming*, vol. 68, pp. 15–47, 1995.

[102] E. Chouzenoux, J.-C. Pesquet, and A. Repetti, "Variable metric forward–backward algorithm for minimizing the sum of a differentiable function and a convex function," *Journal of Optimization Theory and Applications*, vol. 162, no. 1, pp. 107–132, 2014.

[103] P. Frankel, G. Garrigos, and J. Peypouquet, "Splitting methods with variable metric for kurdyka–łojasiewicz functions and general convergence rates," *Journal of Optimization Theory and Applications*, vol. 165, no. 3, pp. 874–900, 2015.

[104] E. Chouzenoux, J.-C. Pesquet, and A. Repetti, "A block coordinate variable metric forward–backward algorithm," *Journal of Global Optimization*, vol. 66, no. 3, pp. 457–485, 2016.

[105] S. Becker and J. Fadili, "A quasi-newton proximal splitting method," *Advances in Neural Information Processing Systems*, vol. 25, 2012.

[106] J. D. Lee, Y. Sun, and M. A. Saunders, "Proximal newton-type methods for minimizing composite functions," *SIAM Journal on Optimization*, vol. 24, no. 3, pp. 1420–1443, 2014.

[107] A. Themelis, L. Stella, and P. Patrinos, "Forward-backward envelope for the sum of two nonconvex functions: Further properties and nonmonotone linesearch algorithms," *SIAM Journal on Optimization*, vol. 28, no. 3, pp. 2274–2303, 2018.

[108] L. Stella, A. Themelis, and P. Patrinos, "Forward–backward quasi-newton methods for nonsmooth optimization problems," *Computational Optimization and Applications*, vol. 67, no. 3, pp. 443–487, 2017.

[109] M. Oudin and J. P. Delmas, "Asymptotic generalized eigenvalue distribution of block multilevel toeplitz matrices," *IEEE Transactions on Signal Processing*, vol. 57, no. 1, pp. 382–387, 2008.

[110] R. B. Lehoucq, D. C. Sorensen, and C. Yang, *ARPACK users' guide: solution of large-scale eigenvalue problems with implicitly restarted Arnoldi methods*. SIAM, 1998.

[111] R. B. Lehoucq, "Implicitly restarted arnoldi methods and subspace iteration," *SIAM Journal on Matrix Analysis and Applications*, vol. 23, no. 2, pp. 551–562, 2001.

[112] D. C. Sorensen, "Implicitly restarted arnoldi/lanczos methods for large scale eigenvalue calculations," in *Parallel Numerical Algorithms*. Springer, 1997, pp. 119–165.

[113] K. Wu and H. Simon, "Thick-restart lanczos method for large symmetric eigenvalue problems," *SIAM Journal on Matrix Analysis and Applications*, vol. 22, no. 2, pp. 602–616, 2000.

[114] G. W. Stewart, "A krylov–schur algorithm for large eigenproblems," *SIAM Journal on Matrix Analysis and Applications*, vol. 23, no. 3, pp. 601–614, 2002.

[115] M. Stoll, "A krylov–schur approach to the truncated svd," *Linear Algebra and its Applications*, vol. 436, no. 8, pp. 2795–2806, 2012.

[116] H. Li and Z. Lin, "Accelerated proximal gradient methods for nonconvex programming," *Advances in Neural Information Processing Systems*, vol. 28, 2015.

[117] E. K. Ryu and W. Yin, "Proximal-proximal-gradient method," *arXiv preprint arXiv:1708.06908*, 2017.

[118] N. Antonello, L. Stella, P. Patrinos, and T. van Waterschoot, "Proximal gradient algorithms: Applications in signal processing," *arXiv preprint arXiv:1803.01621*, 2018.

[119] J. Gillard, "Cadzow's basic algorithm, alternating projections and singular spectrum analysis," *Statistics and its Interface*, vol. 3, no. 3, pp. 335–343, 2010.

[120] C. Lanczos, "An iteration method for the solution of the eigenvalue problem of linear differential and integral operators," *Journal of research of the National Bureau of Standards*, 1950.

[121] M. R. Hestenes, E. Stiefel, *et al.*, "Methods of conjugate gradients for solving linear systems," *Journal of research of the National Bureau of Standards*, vol. 49, no. 6, pp. 409–436, 1952.

[122] A. Chambolle, R. A. De Vore, N.-Y. Lee, and B. J. Lucier, "Nonlinear wavelet image processing: variational problems, compression, and noise removal through wavelet shrinkage," *IEEE Transactions on Image Processing*, vol. 7, no. 3, pp. 319–335, 1998.

[123] I. Daubechies, M. Defrise, and C. De Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, vol. 57, no. 11, pp. 1413–1457, 2004.

[124] J. J. Moré and D. C. Sorensen, "Newton's method," Argonne National Lab., IL (USA), Tech. Rep., 1982.

[125] S. J. Wright, *Numerical optimization*. Springer, 2006.

[126] Y. Park, S. Dhar, S. Boyd, and M. Shah, "Variable metric proximal gradient method with diagonal barzilai-borwein stepsize," in *2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2020, pp. 3597–3601.

[127] C.-L. Wang and C. Li, "A mean value algorithm for toeplitz matrix completion," *Applied Mathematics Letters*, vol. 41, pp. 35–40, 2015.

[128] C.-L. Wang, X.-W. Guo, and X.-H. Yan, "An accelerated proximal gradient algorithm for hankel tensor completion," *Journal of the Operations Research Society of China*, pp. 1–17, 2022.

[129] W. Huang and K. Wei, "Riemannian proximal gradient methods," *Mathematical Programming*, vol. 194, no. 1-2, pp. 371–413, 2022.

[130] H. Zhang and W. W. Hager, "A nonmonotone line search technique and its application to unconstrained optimization," *SIAM journal on Optimization*, vol. 14, no. 4, pp. 1043–1056, 2004.

[131] L. Grippo and M. Sciandrone, "Nonmonotone globalization techniques for the barzilai-borwein gradient method," *Computational Optimization and Applications*, vol. 23, pp. 143–169, 2002.

[132] S. J. Wright, R. D. Nowak, and M. A. Figueiredo, "Sparse reconstruction by separable approximation," *IEEE Transactions on Signal Processing*, vol. 57, no. 7, pp. 2479–2493, 2009.

[133] P. Gong, C. Zhang, Z. Lu, J. Huang, and J. Ye, "A general iterative shrinkage and thresholding algorithm for non-convex regularized optimization problems," in *International Conference on Machine Learning*. PMLR, 2013, pp. 37–45.

[134] Q. Yao, J. T. Kwok, F. Gao, W. Chen, and T.-Y. Liu, "Efficient inexact proximal gradient algorithm for nonconvex problems," *arXiv preprint arXiv:1612.09069*, 2016.

[135] X. Yao and W. Dai, "Accelerated recovery of spectrally sparse signals via modified proximal gradient in hankel space," in *Accepted by IEEE International Conference on Acoustics, Speech, and Signal Processing 2024*, 2024, accepted for presentation.

[136] R. Gu and A. Dogandžić, "Projected nesterov's proximal-gradient algorithm for sparse signal recovery," *IEEE Transactions on Signal Processing*, vol. 65, no. 13, pp. 3510–3525, 2017.

[137] S. Bonettini, I. Loris, F. Porta, and M. Prato, "Variable metric inexact line-search-based methods for nonsmooth optimization," *SIAM journal on optimization*, vol. 26, no. 2, pp. 891–921, 2016.

[138] S. Bonettini, M. Prato, and S. Rebegoldi, "A block coordinate variable metric linesearch based proximal gradient method," *Computational Optimization and Applications*, vol. 71, no. 1, pp. 5–52, 2018.

[139] M. Schmidt, N. Roux, and F. Bach, "Convergence rates of inexact proximal-gradient methods for convex optimization," *Advances in Neural Information Processing Systems*, vol. 24, 2011.

[140] H. Attouch, J. Bolte, and B. F. Svaiter, "Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward–backward splitting, and regularized gauss–seidel methods," *Mathematical Programming*, vol. 137, no. 1, pp. 91–129, 2013.

[141] J. Li, M. S. Andersen, and L. Vandenberghe, "Inexact proximal newton methods for self-concordant functions," *Mathematical Methods of Operations Research*, vol. 85, pp. 19–41, 2017.

[142] S. Becker, J. Fadili, and P. Ochs, "On quasi-newton forward-backward splitting: Proximal calculus and convergence," *SIAM Journal on Optimization*, vol. 29, no. 4, pp. 2445–2481, 2019.

[143] H. Ghanbari and K. Scheinberg, "Proximal quasi-newton methods for regularized convex optimization with linear and accelerated sublinear convergence rates," *Computational Optimization and Applications*, vol. 69, pp. 597–627, 2018.

[144] R. H. Byrd, G. M. Chin, J. Nocedal, and F. Oztoprak, "A family of second-order methods for convex l-1 regularized optimization," *Mathematical Programming*, vol. 159, pp. 435–467, 2016.

[145] R. H. Byrd, J. Nocedal, and F. Oztoprak, "An inexact successive quadratic approximation method for l-1 regularized optimization," *Mathematical Programming*, vol. 157, no. 2, pp. 375–396, 2016.

[146] J. Nocedal and S. J. Wright, *Numerical optimization*. Springer, 1999.

[147] M. Wu, B. Yin, A. Vosoughi, C. Studer, J. R. Cavallaro, and C. Dick, "Approximate matrix inversion for high-throughput data detection in the large-scale mimo uplink," in *ISCAS*. IEEE, 2013, pp. 2155–2158.

[148] X. Yao and D. Wei, "A preconditioned proximal gradient for compressed sensing," in *2023 IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing*, 2023.