Robust Synchronization and 3D Reconstruction from Cryo-EM Images

Lanhui Wang

A Dissertation
Presented to the Faculty
of Princeton University
in Candidacy for the Degree
of Doctor of Philosophy

Recommended for Acceptance
by the Program in
Applied and Computational Mathematics
Adviser: Professor Amit Singer

September 2013
© Copyright by Lanhui Wang, 2013.

All rights reserved.
Abstract

Synchronization of rotations is the problem of estimating a set of rotations \( R_1, R_2, \ldots, R_n \) based on noisy measurements of relative rotations \( R_i^{-1}R_j \). This fundamental problem has found a broad range of applications in computer vision, sensor network localization, structural biology, among others. A closely related problem is the “orientation assignment problem”, which is to estimate the rotations \( R_1, R_2, \ldots, R_n \) of a three-dimensional (3D) object given a finite set of two-dimensional (2D) tomographic projection images \( P_1, P_2, \ldots, P_n \). The “orientation assignment problem” is particularly important in single-particle reconstruction from cryo-electron microscopy (cryo-EM) images, for determining 3D structures of molecules that cannot be crystallized for X-ray.

In this thesis, we propose and analyze new algorithms for the synchronization problem, and the problems on orientation determination and 3D inversion in cryo-EM reconstruction. In the first chapter, we overview the background of the synchronization problem and the problems in cryo-EM reconstruction. In chapter 2 we propose a Least Unsquared Deviations (LUD) method for the synchronization problem, and prove exact and stable recovery of this method for a specific probabilistic model of the measurements. We discover a phase transition behavior of LUD and present a theoretical justification. The LUD problem is solved by semidefinite relaxation (SDR) and the alternating direction method of multipliers (ADMM). In chapter 3 we apply LUD based algorithms to improve the accuracy of orientation determination from cryo-EM images. In chapter 4 we propose a fast and accurate Fourier-based iterative 3D inversion algorithm in cryo-EM, which exploits the Toeplitz structure of the composition of backward and forward projectors. We conclude this thesis in chapter 5 by summarizing our contributions and discussing directions for future work.
Acknowledgements

I would like to express the deepest appreciation to my advisor Professor Amit Singer, who introduced me to the synchronization problem and the problems in cryo-EM. He has inspired me with his scientific vision, enthusiasm, brilliance, and with his tremendous support and help. I feel motivated and encouraged every time we discuss the problems. Without his encouragement and guidance this dissertation would not have been possible.

I would like to thank my committee members, Professor Yoel Shkolnisky and Professor Fred Sigworth for the useful comments, remarks and engagement through the learning process of this thesis. I would like to thank Professor Gilad Lerman for reading and commenting on the manuscript. I would like to thank Professor Zaiwen Wen for introducing me to the ADMM solver for our problems as well for the support on the way.

I would like to thank my committee members for the general examination and the qualifying examination, Professor Ingrid Daubechies, Professor Weinan E and Professor Erhan Cinlar. I am very thankful to all of them for their support and guidance. I thank Professor Pierre-Antoine Absil and Nicolas Boumal for their help while visiting UCL in Belgium, and the stimulating discussions with them. My sincere thanks also goes to Professor Laurent Demanet for offering me the summer internship opportunity in his group.

And finally, I thank the members of my research group: Zhizhen Zhao, Xiuyuan Cheng, Hau-Tieng Wu, Mihai Cucuringu, Afonso Bandeira, Kunal Chaudhury, Yue-haw Khoo, Onur Ozyesil, and Teng Zhang, for their constant support and help.
To my parents.
## Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abstract</td>
<td>iii</td>
</tr>
<tr>
<td>Acknowledgements</td>
<td>iv</td>
</tr>
<tr>
<td>List of Tables</td>
<td>ix</td>
</tr>
<tr>
<td>List of Figures</td>
<td>x</td>
</tr>
</tbody>
</table>

### 1 Introduction

1.1 The Synchronization Problem                                          3

1.2 3D Reconstruction from Cryo-EM images                                5

1.2.1 The Algorithmic Pipeline of Single Particle Reconstruction (SPR)  6

1.2.2 Orientation Determination Using Common-lines Between Images        7

1.2.3 Inversion from Projection Images                                   10

### 2 Exact and Stable Recovery of Rotations for Robust Synchronization  16

2.1 Approximating the Least Squares Solution                             17

2.1.1 Semidefinite Programming Relaxation                               18

2.1.2 Spectral Relaxations                                               21

2.2 Least Unsquared Deviations (LUD) and Semidefinite Relaxation         22

2.3 Exact Recovery of the Gram matrix $G$                                23

2.3.1 Main Theorem                                                       23

2.3.2 Proof of Theorem 2.3.1                                             25

2.3.3 The Value of $c(d)$                                                40
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.3.4 Proof of Lemma 2.3.10</td>
<td>44</td>
</tr>
<tr>
<td>2.3.5 Proof of Lemma 2.3.11</td>
<td>49</td>
</tr>
<tr>
<td>2.4 Stability of LUD</td>
<td>57</td>
</tr>
<tr>
<td>2.4.1 Weak Stability</td>
<td>58</td>
</tr>
<tr>
<td>2.4.2 Strong Stability</td>
<td>62</td>
</tr>
<tr>
<td>2.5 A Generalization of LUD to Random Incomplete Measurement Graphs</td>
<td>67</td>
</tr>
<tr>
<td>2.6 Alternating Direction Augmented Lagrangian Method of Multipliers</td>
<td>68</td>
</tr>
<tr>
<td>2.7 Numerical Experiments</td>
<td>73</td>
</tr>
<tr>
<td>2.7.1 Experiments with Full Measurements</td>
<td>74</td>
</tr>
<tr>
<td>2.7.2 Experiments with Incomplete Measurements</td>
<td>81</td>
</tr>
<tr>
<td>2.7.3 Real Data Experiments</td>
<td>82</td>
</tr>
<tr>
<td>2.8 Summary</td>
<td>85</td>
</tr>
<tr>
<td>3 Orientation Determination of Cryo-EM images Using Least Un-squared Deviations</td>
<td>89</td>
</tr>
<tr>
<td>3.1 Detection of Common Lines Between Images</td>
<td>90</td>
</tr>
<tr>
<td>3.2 Weighted LS and LUD</td>
<td>92</td>
</tr>
<tr>
<td>3.3 Semidefinite Programming Relaxation (SDR) and the Rounding Procedure</td>
<td>94</td>
</tr>
<tr>
<td>3.3.1 Constructing the Gram matrix $\tilde{G}$ from the Rotations $R_i$</td>
<td>95</td>
</tr>
<tr>
<td>3.3.2 SDR for Weighted LS</td>
<td>96</td>
</tr>
<tr>
<td>3.3.3 SDR for LUD</td>
<td>97</td>
</tr>
<tr>
<td>3.3.4 The Randomized Rounding Procedure</td>
<td>98</td>
</tr>
<tr>
<td>3.3.5 Exact Recovery of the Gram matrix $\tilde{G}$ from Correct Common-lines</td>
<td>100</td>
</tr>
<tr>
<td>3.4 The Spectral Norm Constraint</td>
<td>102</td>
</tr>
<tr>
<td>Section</td>
<td>Title</td>
</tr>
<tr>
<td>---------</td>
<td>-------</td>
</tr>
<tr>
<td>3.5</td>
<td>The Alternating Direction Method of Multipliers (ADMM) for SDRs</td>
</tr>
<tr>
<td>3.5.1</td>
<td>The Relaxed Weighted LS Problem</td>
</tr>
<tr>
<td>3.5.2</td>
<td>The Relaxed LUD problem</td>
</tr>
<tr>
<td>3.6</td>
<td>The Iterative Reweighted Least Squares (IRLS) Procedure</td>
</tr>
<tr>
<td>3.7</td>
<td>Numerical Results</td>
</tr>
<tr>
<td>3.7.1</td>
<td>Experiments on Simulated Images</td>
</tr>
<tr>
<td>3.7.2</td>
<td>Experiments on a Real Dataset</td>
</tr>
<tr>
<td>3.8</td>
<td>Discussion</td>
</tr>
<tr>
<td>4</td>
<td>A Fourier-based Approach for Inversion from Cryo-EM Images</td>
</tr>
<tr>
<td>4.1</td>
<td>Preliminaries</td>
</tr>
<tr>
<td>4.1.1</td>
<td>Notation</td>
</tr>
<tr>
<td>4.1.2</td>
<td>Fourier Transform Conventions</td>
</tr>
<tr>
<td>4.1.3</td>
<td>Toeplitz Matrices and Circulant Matrices</td>
</tr>
<tr>
<td>4.2</td>
<td>A Fourier-based Approach for 3D Inversion</td>
</tr>
<tr>
<td>4.2.1</td>
<td>The Forward-Projector $A$</td>
</tr>
<tr>
<td>4.2.2</td>
<td>The Back-projector $A^*$ and the Toeplitz Structure of $A^*A$</td>
</tr>
<tr>
<td>4.2.3</td>
<td>The Conjugate Gradient (CG) Method</td>
</tr>
<tr>
<td>4.2.4</td>
<td>Fourier-based Iterative Reconstruction Method (FIRM)</td>
</tr>
<tr>
<td>4.3</td>
<td>Numerical Results</td>
</tr>
<tr>
<td>4.4</td>
<td>Summary and Discussion</td>
</tr>
<tr>
<td>5</td>
<td>Conclusion and Future Work</td>
</tr>
<tr>
<td>Bibliography</td>
<td></td>
</tr>
</tbody>
</table>
# List of Tables

2.1 The Relative Error of the Gram matrix $\hat{G}$ obtained by LUD. 75

2.2 MSEs of the estimated rotations in $SO(2)$ using EIG, SDP and LUD. 76

2.3 MSEs of the estimated rotations in $SO(3)$ using EIG, SDP and LUD. 77

3.1 The average cost time using different algorithms in the problem of orientation determination from cryo-EM images. 128
List of Figures

1.1 Fourier projection-slice theorem and common-lines. ........................ 8
1.2 Fourier projection-slice theorem and the inverse problem in cryo-EM. 12
1.3 Three CTFs corresponding to different defocus values. ................. 13

2.1 Two \( n \times n \) matrices ............................................ 52
2.2 Phase transitions of LUD on complete graphs. ......................... 78
2.3 Stability of LUD on complete graphs. ................................ 80
2.4 The linear relationship of \( \log(\text{MSE}) \) and \( \log(\kappa) \) in robust synchronization, where \( \kappa \) represents the level of perturbations on the “good” edges. 80
2.5 Phase transitions of LUD on incomplete graphs. .................... 81
2.6 Stability of LUD on incomplete graphs. ................................ 82
2.7 The Lucy statue. ...................................................... 83
2.8 Histogram of the unsquared residuals of EIG, SDP and LUD for the Lucy dataset. ...................................................... 83
2.9 True and estimated rotation ratio measurement errors for the Lucy dataset. ...................................................... 84
2.10 Front and back views of the reconstructions from the Lucy dataset using LUD and EIG. ...................................................... 85

3.1 Outliers in the detected common-lines and the difference between the squared distance and the absolute deviation. ...................... 93
<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.2</td>
<td>The dependency of the spectral norm of $\tilde{G}$ on the distribution of orientations of the cryo-EM images</td>
</tr>
<tr>
<td>3.3</td>
<td>Simulated cryo-EM images</td>
</tr>
<tr>
<td>3.4</td>
<td>Reconstructed volumes from simulated cryo-EM images</td>
</tr>
<tr>
<td>3.5</td>
<td>FSCs of the reconstructed volumes from the simulated cryo-EM images</td>
</tr>
<tr>
<td>3.6</td>
<td>Real cryo-EM images and noise reduction by image averaging</td>
</tr>
<tr>
<td>3.7</td>
<td>Initial models and refined models reconstructed from real cryo-EM images</td>
</tr>
<tr>
<td>3.8</td>
<td>Convergence of the refinement process</td>
</tr>
<tr>
<td>4.1</td>
<td>Plot of residual norm versus iteration count for a typical CG iterative 3D inversion from images</td>
</tr>
<tr>
<td>4.2</td>
<td>FSCs of the reconstructed volumes excluding or within the missing cone region</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

Finding a global assignment of rotations $R_1, R_2, ..., R_n$ in the special orthogonal group $SO(d)$ from noisy measurements of a subset of their pairwise ratios $R_i^{-1}R_j$ is known as the synchronization problem. The problem has received considerable attention in recent years due to its importance in applications in computer vision, computer graphics, sensor network localization, structural biology and more. For example, the angular embedding for image reconstruction from pairwise intensity differences is a synchronization problem in $SO(2)$, and the global alignment of multiple 3D scans from different single views in computer graphics is a synchronization problem over $SO(3)$.

The problem of orientation determination of cryo-EM images is closely related to the synchronization problem. Cryo-EM is used to acquire 2D projection images of thousands of individual, identical frozen-hydrated macromolecules at random unknown orientations and positions. The goal is to reconstruct the 3D structures of macromolecules with sufficiently high resolution such that fine details can be interpreted. However, the collected images are extremely noisy due to the limited electron dose used for imaging to avoid excessive beam damage. In addition, the pose parameters (orientations and positions) of the imaged particles are unknown. Thus
estimation of the orientations $R_1, R_2, ..., R_n$ of cryo-EM images is important in the 3D reconstruction procedure. Unlike the synchronization problem, the pairwise information given here is not a set of relative rotations $R_i^{-1}R_j$; instead, it is a set of so-called “common-lines” between cryo-EM images. The goal is to estimate orientations that will properly align all the common-lines between cryo-EM images in a globally consistent way. Both of the synchronization problem and the problem of orientation determination of cryo-EM images can be solved by minimizing a penalty function over the non-convex set $SO(d) \times \ldots \times SO(d)$.

After the orientation determination of cryo-EM images, we turn to obtain the 3D structure of macromolecules by solving a linear inverse problem corresponding to the set of 2D projection images. The inverse problem is difficult due to the high level of noise in the cryo-EM images and limiting effects of the contrast transfer functions of the electron microscope. In addition, the inverse problem is ill-conditioned, due to the geometry of the problem and possible incomplete coverage of the viewing directions. The inversion takes place in the ab-inito model reconstruction and each iteration of the refinement process. Thus an inversion algorithm must be fast and accurate, in order to accelerate the iterative refinement process and to reduce the propagation of inversion errors.

This chapter is organized as follows. Section 1.1 presents an overview of the synchronization problem and a brief introduction to our work on Least Unsquared Deviations (LUD) method. In section 1.2 the background of cryo-EM is firstly introduced. Following this introduction, section 1.2.1 overviews the algorithmic pipeline of SPR in cryo-EM. Section 1.2.2 presents the related work on the problem of orientation determination of cryo-EM images, and the main ideas of our LUD based algorithms. Section 1.2.3 introduces the previous work on the inverse problem in cryo-EM, and our work on Fourier-based Iterative Reconstruction Method (FIRM).
1.1 The Synchronization Problem

The synchronization problem over the special orthogonal group $SO(d)$ of rotations in $\mathbb{R}^d$

$$SO(d) = \{ R \in \mathbb{R}^{d \times d} : R^T R = RR^T = I_d, \ \det R = 1 \} \quad (1.1)$$

consists of estimating a set of $n$ rotations $R_1, \ldots, R_n \in SO(d)$ from a subset of (perhaps noisy) measurements $R_{ij}$ of their ratios $R_i^{-1} R_j$. The subset of available ratio measurements is viewed as the edge set of an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, with $|\mathcal{V}| = n$. The goal is to find $R_1, \ldots, R_n$ that satisfy

$$R_i^{-1} R_j \approx R_{ij}, \quad \text{for } (i, j) \in \mathcal{E}. \quad (1.2)$$

Synchronization over the rotation group $SO(d)$ has many applications. Synchronization over $SO(2)$ plays a major role in the framework of angular embedding for ranking and for image reconstruction from pairwise intensity differences [111, 112] and for a certain algorithm for sensor network localization [13]. Synchronization over $SO(3)$ is invoked by many algorithms for structure from motion in computer vision [54, 97, 38, 3], by algorithms for global alignment of 3-D scans in computer graphics [98], and by algorithms for finding 3-D structures of molecules using NMR spectroscopy [15] and cryo-electron microscopy [83, 88]. A closely related problem in terms of applications and methods is the synchronization over the orthogonal group $O(d)$, where the requirement of positive determinant in (1.1) is alleviated. We remark that the algorithms and analysis presented in chapter 2 follow seamlessly to the case of $O(d)$. We choose to focus on $SO(d)$ only because this group is encountered more often in applications.

If the measurements $R_{ij}$ are noiseless and the graph $\mathcal{G}$ is connected then the synchronization problem can be easily solved by considering a spanning tree in $\mathcal{G}$, setting the rotation of the root node arbitrarily, and determining all other rotations.
by traversing the tree while sequentially multiplying the rotation ratios. The rotations obtained in this manner are uniquely determined up to a global rotation, which is the intrinsic degree of freedom of the synchronization problem. However, when the ratio measurements are corrupted by noise, the spanning tree method suffers from accumulation of errors. Estimation methods that use all available ratio measurements and exploit the redundancy of graph cycles are expected to perform better.

If some (though possibly not all) pairwise measurements are noise-free, then a cycle-based algorithm can be used to find the noise-free ratio measurements. Specifically, in order to determine if a ratio measurement is “good” (noise-free) or “bad” (corrupted by random noise), one can examine cycles in the graph that include that edge and check their consistency. A consistent cycle is a cycle for which sequentially multiplying the rotation ratios along the cycle results in the identity rotation. Under the random noise assumption, ratio measurements along consistent cycles are almost surely “good”, and if the subgraph associated with the “good” measurements is connected, then the spanning tree method can be used to determine the rotations. However, cycle-based algorithms have two main weaknesses. First, the computational complexity of cycle-based algorithms increases exponentially with the cycle length. Second, the cycle-based algorithms are unstable to small perturbations on the “good” ratio measurements.

Methods that are based on least squares have been proposed and analyzed in the literature. While the resulting problem is non-convex, the solution to the least squares problem is approximated by either a spectral relaxation (i.e., using leading eigenvectors) or by semidefinite programming (SDP) (see, e.g., [84, 43, 111, 6]). Typically in applications, the ratio measurements generally consist of noisy inliers, which are explained well by the rotations $R_1, \ldots, R_n$, along with outliers, that have no structure. The least squares method is sensitive to these outliers.
In chapter 2, we propose to estimate the rotations by minimizing a different, more robust self consistency error, which is the sum of unsquared residuals [62, 92, 47], rather than the sum of squared residuals. The minimization problem is semidefinite relaxed and solved by the alternating direction method. Moreover, we prove that under some conditions the rotations $R_1, \ldots, R_n$ can be exactly and stably recovered (up to a global rotation), see Theorems 2.3.1, 2.4.1 and 2.4.4 for the complete graph, and Theorems 2.5.1, 2.5.3 for random graphs. Our numerical experiments demonstrate that the new method significantly improves the estimation of rotations, and in particular, achieving state-of-the-art results.

1.2 3D Reconstruction from Cryo-EM images

Single particle reconstruction (SPR) from cryo-electron microscopy (cryo-EM) [25, 103] is an emerging technique for determining the three-dimensional structure of macromolecules. One of the main challenges in SPR is to attain a resolution of 4 Å or better, thereby allowing interpretation of atomic coordinates of macromolecular maps [27, 107]. Although X-ray crystallography and NMR spectroscopy can achieve higher resolution levels ($\sim 1$ Å by X-ray crystallography and 2-5 Å by NMR spectroscopy), these traditional methods are often limited to relatively small molecules. In contrast, cryo-EM is typically applied to large molecules or assemblies with size ranging from 10 to 150 nm, such as ribosomes [26], protein complexes, and viruses.

Cryo-EM is used to acquire 2D projection images of thousands of individual, identical frozen-hydrated macromolecules at random unknown orientations and positions. The collected images are extremely noisy due to the limited electron dose used for imaging to avoid excessive beam damage. A 3D electron density map can be reconstructed from the 2D projection images computationally. However, a broad computational framework is required for the reconstruction.
1.2.1 The Algorithmic Pipeline of Single Particle Reconstruction (SPR)

The main steps of a full SPR procedure can be summarized as follows.

1. Particle selection: the individual molecules are detected and extracted from the acquired data.

2. Class averaging: images with similar projection directions are averaged together to improve their SNR.

3. Orientation estimation: this is done via the random-conical tilt technique, or common-lines based approaches, or reprojection. This step may also include a shift estimation.

4. Reconstruction: a 3D image is generated by a tomographic inversion algorithm.

5. Iterative refinement: the whole procedure is repeated from step 3, using the outcome of the most recent reconstruction to improve the results.

Numerous technical and computational advances have contributed to the different processing steps in SPR. First, the particles can be automatically selected from micro-graphs using algorithms based on template matching, computer vision, neural network, or others [118]. Then multivariate statistical data compression [46] [102] and classification techniques [100] [66] [88] can be used to sort and partition the large set of particle images by their viewing directions, producing “class averages” of enhanced signal-to-noise ratio (SNR). After class averaging, an ab-initio estimation of the pose parameters can be obtained using the random-conical tilt technique [75] or common-lines based approaches [101] [85] [86]. Using the ab-initio estimation of the pose parameters, a preliminary 3D map is reconstructed from the projection images by a tomographic inversion algorithm. The tomographic inversion is either computed
by applying weighted adjoint operator \[75, 74, 37, 67\], or by using an iterative approach for inversion \[31, 53, 28\]. The initial model is then iteratively refined \[65\] in order to obtain a higher-resolution 3D reconstruction. In each iteration of the refinement process, the current 3D model is projected at several pre-chosen viewing directions and the resulting reprojection images are matched with the particle images, giving rise to new estimates of their pose parameters. The new pose parameters are then used to produce a refined 3D model using a 3D reconstruction algorithm. This process is repeated for several iterations until convergence.

However, due to the intrinsic poor SNR in cryo-EM images and the increasing number of particles, the 3D reconstruction processing remains several computational challenges, which come from the complexity of the algorithms themselves (e.g. classification, orientation determination and 3D inversion). In this thesis, we address the problem of orientation estimation using common-lines between cryo-EM images and the inverse problem in 3D reconstruction, which are introduced in section \[1.2.2\] and section \[1.2.3\] respectively.

### 1.2.2 Orientation Determination Using Common-lines Between Images

In the common-lines based reconstruction methods, the Fourier projection-slice theorem (see, e.g., \[89\]) plays a fundamental role. The theorem states that restricting the 3D Fourier transform of the volume to a planar central slice yields the Fourier transform of a 2D projection of the volume in a direction perpendicular to the slice (Figure \[1.1\]). Thus, any two projections imaged from non-parallel viewing directions intersect at a line in Fourier space, which is called the common-line between the two images. The common-lines between any three images with linearly independent projection directions determine their relative orientation up to handedness. This is the basis of the “angular reconstitution” technique of van Heel \[101\], which was also
Figure 1.1: Fourier projection-slice theorem. In the middle, $\hat{P}_i$ is a polar Fourier transform of projection $P_i$ on the left. The red line $\vec{c}_{ij}$ represents the direction of a common-line between $\hat{P}_i$ and $\hat{P}_j$ on $\hat{P}_i$. On the right, the two transformed images $\hat{P}_i$ and $\hat{P}_j$ intersect with each other at the common-line after rotations $R_i$ and $R_j$, yielding the equation (3.4).

developed independently by Vainshtein and Goncharov [99]. In this technique, the orientations of additional projections are determined in a sequential manner. Farrow and Ottensmeyer [22] used quaternions to obtain the relative orientation of a new projection in a least square sense. The main problem with such techniques is that they are sensitive to false detection of common lines that leads to the accumulation of errors. Penczek et.al. [71] tried to obtain the rotations corresponding to all projections simultaneously by minimizing a global energy functional, which requires a brute force search in an exponentially large parametric space of all possible orientations for all projections. Mallick et. al. [51] and Singer et al. [85] applied Bayesian approaches to use common-lines information from different groups of projections. Recently, Singer and Shkolnisky [86] developed two algorithms based on eigenvectors and semidefinite programming for estimating the orientations of all images. These two algorithms correspond to convex relaxations of the global self-consistency error minimization, and can accurately estimate all orientations at relatively low common-line detection rates.
When the signal-to-noise ratio (SNR) of the image is significantly low, the detected common-lines consist of a modest number of noisy inliers, which are explained well by the image orientations, along with a large number of outliers, that have no structure. The standard common-lines based methods, including those using least squares (LS) \[22, 86\], are sensitive to these outliers. In chapter 3 we estimate the orientations using a different, more robust self-consistency error, which is the sum of unsquared residuals \[62, 92\], rather than the sum of squared residuals of the LS formulation. Convex relaxations of least unsquared deviations (LUD) have been recently proposed for other applications, such as robust principal component analysis \[47\] and robust synchronization of orthogonal transformations \[108\]. Under certain noise models for the distribution of the outliers (e.g., the haystack model of \[47\]), such convex relaxations enjoy proven guarantees for exact and stable recovery with high probability. Such theoretical and empirical improvements that LUD brings compared to LS serve as the main motivation to consider in chapter 3 the application of LUD to the problem of orientation estimation from common-lines in single particle reconstruction.

The LUD minimization problem is solved here via semidefinite relaxation. When the detection rate of common-lines is extremely low, the estimated viewing directions of the projection images are observed to cluster together. This artificial clustering can be explained by the fact that images that share the same viewing direction also share more than one common line. In order to mitigate this spurious clustering of estimated viewing directions, we add to the minimization formulation a spectral norm term, either as a constraint or as a regularization term. The resulting minimization problem is solved by the alternating direction method of multipliers (ADMM), which has been proved to converge to the global minimizer in many cases \[42\]. We also consider the application of the iteratively reweighted least squares (IRLS) procedure, which is not guaranteed to converge to the global minimizer, but performs well in our
numerical experiments. We demonstrate that the ab-initio models resulted by our new methods are more accurate and require fewer refinement iterations compared to least squares based methods.

1.2.3 Inversion from Projection Images

After the orientations of cryo-EM images are determined, we consider the problem of inverting the set of 2D projection images to obtain 3D structures of the macromolecules. The Fourier projection-slice theorem also plays a fundamental role in the inversion algorithms independent of whether they are implemented in real space or in Fourier space \[103\] (Figure 1.2). It follows from the theorem that a reconstruction can be obtained by a 3D inverse Fourier transform from the Fourier domain which is filled in by the 2D Fourier slices. Although the continuous Fourier transform is a unitary linear transformation whose inverse equals its adjoint, the 3D discrete inverse Fourier transform of the slices does not equal its adjoint due to the non-uniform sampling in the frequency domain. Observe that the 3D Fourier space filled by 2D slices is denser at low frequencies and sparser at high frequencies. As a result, when the adjoint operator is applied to the slices, the low-frequency information of the macromolecule is overemphasized compared to the high frequency information, meaning that the inverse problem cannot be simply solved in this way. Instead, the solution to the linear inverse problem is either computed by applying a carefully designed weighted adjoint operator that addresses the non-uniform sampling \[75, 74, 37, 67\], or by using an iterative approach for inversion \[31, 53, 28\].

Many techniques have been developed to reconstruct a volume from projection images \[64\]. The Algebraic Reconstruction Technique (ART) \[31, 53\] and the Simultaneous Iterative Reconstruction Technique (SIRT) \[28\] are algebraic approaches to find a 3D reconstruction such that its 2D re-projections are most similar to the input images in the least squares sense. The results of ART and SIRT are very
accurate and they can incorporate additional constraints for the volume according to possible prior knowledge, such as positivity and smoothness. Another important advantage of ART and SIRT is that they are able to reconstruct from projection images with unevenly distributed viewing directions, which is usually the situation in cryo-EM since the macromolecules can assume various stable positions depending on their shape and adsorption properties [103, 25]. However, ART and SIRT are extremely time-consuming if many iterations are needed for convergence. The filtered back-projection approaches, including the general weighted back-projection with exponent-based weighting function (WBP1) [73, 74] and the exact filter weighted back-projection (WBP2) [37] are considerably faster. However, the suitability of their weighting schemes depends on the distribution of the viewing directions, which can affect the precision of the reconstruction. The Gridding Direct Fourier Reconstruction (GDFR) [67] is a relatively recent reconstruction technique. During the preprocessing stage, GDFR re-samples the 2D central slices onto 1D central radial lines to form a special structure of a non-uniform grid. Then the gridding weights are computed via a spherical Voronoi diagram. Finally, with the gridding weights, the numerical inverse Fourier transform is computed by the 3D gridding method. Although GDFR is both accurate and fast, it is limited to cases when there is no major gap among the viewing directions of the images since the proper gridding weights depend on the full coverage of Fourier space by the Fourier slices. Another direct Fourier inversion algorithm is the nearest neighbor direct inversion reconstruction algorithm (4NN) [114, 69]. In the algorithm of 4NN, the 2D projections are first padded with zeros to four times the size, 2D Fourier transformed, and samples are accumulated within the target 3D Fourier volume using simple nearest neighbor interpolation. In the process, a 3D weighting function modeled on Bracewells local density [9] is constructed and applied to individual voxels of 3D Fourier space to account for possible non-uniform distribution of samples. 4NN is even faster than GDFR and it is accurate when the
Figure 1.2: The Fourier projection-slice theorem states that a slice extracted from the frequency domain representation of a 3D volume yields the Fourier transform of a projection of the volume in a direction perpendicular to the slice. The volume we show here is the 50S ribosomal subunit used in our numerical experiments. The molecular surface was produced using the UCSF Chimera package \cite{72} from the Resource for Biocomputing, Visualization, and Informatics at the University of California, San Francisco (supported by NIH P41 RR001081).

Sampling points are uneven in Fourier space. However, 4NN cannot avoid the projections whose Fourier transforms are close to gaps in Fourier space from receiving excessive weight.

Cryo-EM images, however, are not merely 2D mathematical projections of the macromolecule. During the imaging process, the objective lens of the electron microscope imposes a contrast transfer function (CTF) on a group of images \cite{25}. A CTF is approximated by a sinusoidal function in Fourier space depending on the magnitude of the frequency (Figure \ref{13}), and it is also possible to improve the estimation of the CTF from the cryo-EM images themselves \cite{117, 44}. The CTF affects the acquired images through multiplication in the 2D Fourier domain, or equivalently, through a convolution in the real domain. The CTFs modulate the Fourier transform of true projections in a defocus-dependent way. A group of images taken using the same defocus setting is called a defocus group. One generally works at relatively large de-
Figure 1.3: Three CTFs corresponding to different defocus values. The CTFs are generated according to the defocus formula from page 81 of [44]. Parameters: $\alpha = 0.07$ (amplitude contrast component), the electron wavelength $\lambda = 2.51$ picometers, Cs=2.0 (spherical aberration constant), B-factor=100, and defocus=1.4µm, 1.75µm, and 2µm. Pixel size is 3.36Å.

To reconstruct an undistorted volume from different defocus groups, one must carry out CTF correction of images or volumes. In the defocus groups approach, 3D CTF correction is applied to the reconstructed volume from each defocus group, then these volumes are combined to form a single, CTF-corrected volume [68, 117]. Another approach is to use CTF-corrected images or class averages for reconstruction. These two approaches take reconstruction and CTF-correction as two separate steps. In the step of reconstruction, one of the reconstruction algorithms (e.g. SIRT, WBP1, WBP2, GDFR, etc.) is used. In the step of CTF-correction, the Wiener filter is applied to find the least square solution to the problem of CTF-correction.
In contrast with these two approaches, it is possible to incorporate CTF corrections into the reconstruction algorithms in hope of a better merging of different defocus groups. In [68], Penczek et al. describe an algebraic method in real space to find a 3D reconstruction such that its 2D reprojections with CTFs are most similar to the input images in the least squares sense. Penczek et al. conclude that this approach outperforms the defocus groups approach. However, the algebraic method is time consuming. A similar idea is used in the refinement process by FREALIGN [33], which creates a reconstruction by computing a least-squares fit to all the images with weights depending on the CTFs and the correlations between the images and the references. However, this reconstruction method is limited to the refinement process and cannot be applied for reconstructing an initial model. The direct Fourier inversion algorithm 4NN [114, 69] efficiently incorporates CTF corrections during the nearest neighbor interpolation using the Wiener filter methodology.

In chapter 4, we propose a fast and accurate Fourier-based Iterative Reconstruction Method (FIRM). FIRM is faster than ART and SIRT while maintaining their advantages over WBP1, WBP2, GDFR and 4NN: the results of FIRM are very accurate, it can incorporate prior knowledge, and does not require the viewing directions of the images to be evenly sampled. In addition, the time complexity of FIRM has the same order of magnitude as the fast algorithm 4NN and the actual running time differs only by a constant factor of around 5. Moreover, FIRM is also flexible to incorporate CTF corrections.

To derive the FIRM algorithm, we use a forward projecting model \( \mathbf{b} = \mathbf{A} (\mathbf{V}) + \text{noise} \) according to the Fourier projection-slice theorem, where \( \mathbf{A} \) is the forward projector, \( \mathbf{V} \) is the unknown 3D density map of the macromolecule we are interested in, and \( \mathbf{b} \) is the 2D Fourier transform of the noisy images. The reconstruction is obtained by minimizing the cost function \( \rho (\mathbf{V}) = \| \mathbf{b} - \mathbf{A} (\mathbf{V}) \|^2 \). FIRM takes advantage of the Toeplitz structure of \( \mathbf{A}^* \mathbf{A} \), where \( \mathbf{A}^* \) is the adjoint operators of \( \mathbf{A} \). The Toeplitz
structure of the composition of the backward and forward projectors has already been successfully used for 2D reconstruction of an image from non-uniform Fourier-domain samples \cite{18, 34, 106, 23}. We extend the usage of the Toeplitz structure to address the 3D reconstruction problem. Due to the Toeplitz structure of the operator $A^*A$, it is equivalent to a convolution with a kernel. The kernel is precomputed using the non-uniform Fast Fourier Transform (NUFFT) \cite{21, 32, 24} and is efficiently applied in an iterative process, such as the Conjugate Gradient (CG) method, to estimate the 3D map $V$. 
Chapter 2

Exact and Stable Recovery of Rotations for Robust Synchronization

The synchronization problem over the special orthogonal group $SO(d)$ consists of estimating a set of unknown rotations $R_1, R_2, \ldots, R_n$ from noisy measurements of a subset of their pairwise ratios $R_i^{-1}R_j$. Synchronization of rotations appears naturally in a number of important applications such as computer vision, computer graphics, and sensor network localization. In this chapter, we describe the least unsquared deviations (LUD) method for robust synchronization, which can achieve exact and stable recovery of rotations. In section 2.1, we review existing SDP and spectral relaxation methods for approximating the least squares solution. In section 2.2, we derive the more robust LUD cost function and its convex relaxation. In section 2.3, we introduce the noise model and prove conditions for exact recovery by the LUD method. In section 2.4, we prove that the recovery of the rotations is stable to noise. In section 2.5, we generalize the results to the case of random (incomplete) measurement graphs. In section 2.6, we discuss the application of the alternating
direction method of multipliers for solving the LUD optimization problem. The results of numerical experiments on both synthetic data as well as for global alignment of 3D scans are reported in section 2.7. Finally, section 2.8 is a summary.

2.1 Approximating the Least Squares Solution

In this section we overview existing relaxation methods that attempt to approximate the least squares solution. In the synchronization problem, the set of available measurements gives rise to a graph structure, where the \( n \) nodes correspond to the rotations \( R_i \) and an edge is present between two nodes \( i \) and \( j \) if a measurement of \( R_i^{-1}R_j \) is given. The goal is to estimate the set of rotations \( R_1, \ldots, R_n \), given the measurement graph. The least squares solution is the set of rotations that minimize the sum of squared deviations

\[
\min_{R_1, \ldots, R_n \in SO(d)} \sum_{(i,j) \in \mathcal{E}} w_{ij} \left\| R_i^{-1}R_j - R_{ij} \right\|^2,
\]

where \( \| \cdot \| \) denotes the Frobenius norm\(^1\), \( w_{ij} \) are non-negative weights that reflect the measurement precisions\(^2\), \( R_{ij} \) are the noisy measurements of rotation ratios \( R_i^{-1}R_j \), and \( \mathcal{E} \) is the index set of the measurements. The feasible set \( SO(d)^n = SO(d) \times \cdots \times SO(d) \) of the minimization problem (2.1) is however non-convex. Convex relaxations of (2.1) involving SDP and spectral methods have been previously proposed and analyzed.

\(^1\)The Frobenius norm of an \( m \times n \) matrix \( A \) is defined as \( \| A \| = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} A_{ij}^2} \).

\(^2\)If all measurements have the same precision, then the weights take the simple form \( w_{ij} = 1 \) for \( (i,j) \in \mathcal{E} \) and \( w_{ij} = 0 \) otherwise.
2.1.1 Semidefinite Programming Relaxation

Convex relaxation using SDP was introduced in [84] for $SO(2)$ and in [3] for $SO(3)$ and is easily generalized for any $d$. The method draws similarities with the Goemans and Williamson approximation algorithm to Max-Cut that uses SDP [30].

The first step of the relaxation involves the observation that the least squares problem (2.1) is equivalent to the maximization problem

$$\max_{R_1,\ldots,R_n \in SO(d)} \sum_{(i,j) \in E} w_{ij} \operatorname{Tr}(R_i^{-1}R_jR_j^T),$$

(2.2)

due to the fact that $\|R_i^{-1}R_j\|^2 = \|R_{ij}\|^2 = d$.

The second step of the relaxation introduces the matrix $G$ of size $n \times n$ whose entries

$$G_{ij} = R_i^T R_j$$

(2.3)

are themselves matrices of size $d \times d$, so that the overall size of $G$ is $nd \times nd$. The matrix $G$ admits the decomposition

$$G = R^T R,$$

(2.4)

where $R$ is a matrix of size $d \times nd$ given by

$$R = \begin{bmatrix} R_1 & R_2 & \cdots & R_n \end{bmatrix}.$$  

(2.5)

The objective function in (2.2) can be written as $\operatorname{Tr}(GC)$, where the entries of $C$ are given by $C_{ij} = w_{ij}R_i^T R_j$ (notice that $C$ is symmetric, since $R_i^T = R_i$ and $w_{ij} = w_{ji}$).

The matrix $G$ has the following properties:

1. $G \succeq 0$, i.e., it is positive semidefinite (PSD).
2. $G_{ii} = I_d$ for $i = 1,\ldots,n$. 

18
3. \( \text{rank}(G) = d. \)

4. \( \det(G_{ij}) = 1 \) for \( i, j = 1, \ldots, n. \)

Relaxing properties (3) and (4) leads to the following SDP:

\[
\max_G \text{Tr}(GC) \quad \text{(2.6)}
\]

s.t. \( G \succeq 0 \)

\[ G_{ii} = I_d, \quad \text{for } i = 1, 2, \ldots, n. \]

Notice that for \( d = 1 \) (2.6) reduces to the SDP for MAX-CUT [30]. Indeed, synchronization over \( O(1) \cong \mathbb{Z}_2 \) is equivalent to MAX-CUT. \(^3\)

The third and last step of the relaxation involves the rounding procedure. The rotations \( R_1, \ldots, R_n \) need to be obtained from the solution \( G \) to (2.6). The rounding procedure can be either random or deterministic. In the random procedure, a matrix \( Q \) of size \( nd \times d \) is sampled from the uniform distribution over matrices with \( d \) orthonormal columns in \( \mathbb{R}^{nd} \) (see [55] for description of the sampling procedure).

The Cholesky decomposition of \( G = LL^T \) is computed, and the product \( LQ \) of size \( nd \times d \) is formed, viewed as \( n \) matrices of size \( d \times d \), denoted by \( (LQ)_1, \ldots, (LQ)_n \).

The estimate for the inverse of the \( i \)'th rotation \( \hat{R}_i^T \) is then obtained via the singular value decomposition (SVD) (equivalently, via the polar decomposition) of \( (LQ)_i \) as (see, e.g., [40, 56, 79])

\[
(LQ)_i = U_i \Sigma_i V_i^T, \quad J = \begin{bmatrix} I_{d-1} & 0 \\ 0 & \det U_i V_i^T \end{bmatrix}, \quad \hat{R}_i^T = U_i J V_i^T. \quad \text{(2.7)}
\]

---

\(^3\)The case of \( SO(2) \) is special in the sense that it is possible to represent group elements as complex-valued numbers, rendering \( G \) a complex-valued Hermitian positive semidefinite matrix of size \( n \times n \) (instead of a \( 2n \times 2n \) real valued positive semidefinite matrix).
(the only difference for synchronization over $O(d)$ is that $\hat{R}_i^T = U_i V_i^T$, excluding any usage of $J$). In the deterministic procedure, the top $d$ eigenvectors $v_1, \ldots, v_d \in \mathbb{R}^{nd}$ of $G$ corresponding to its $d$ largest eigenvalues are computed. A matrix $T$ of size $nd \times d$ whose columns are the eigenvectors is then formed, i.e., $T = \begin{bmatrix} v_1 & \cdots & v_d \end{bmatrix}$.

As before, the matrix $T$ is viewed as $n$ matrices of size $d \times d$, denoted $T_1, \ldots, T_n$ and the SVD procedure detailed in (2.7) is applied to each $T_i$ (instead of $(GQ)_i$) to obtain $R_i$.

We remark that the least squares formulation (2.1) for synchronization of rotations is an instance of quadratic optimization problems under orthogonality constraints (QP-Oc) [60, 90]. Other applications of QP-Oc are the generalized orthogonal Procrustes problem and the quadratic assignment problem. Different semidefinite relaxations for the Procrustes problem have been suggested in the literature [60, 58, 90].

In [60], for the problem (2.2), the orthogonal constraints $R_i^T R_i = I_d$ can be relaxed to $\|R_i\| \leq 1$, and the resulting problem can be converted to a semidefinite program with one semidefinite constraint for a matrix of size $nd^2 \times nd^2$, $2n$ semidefinite constraints from matrices of size $d \times d$ (see (55) in [60]) and some linear constraints. Thus, compared with that relaxation, the one we use in (2.6) has a lower complexity, since the problem (2.6) has size $nd \times nd$. As for the approximation ratio of the relaxation, if $C$ is positive semidefinite (as in the case of the Procrustes problem), then there is a constant approximation ratio for the relaxation (2.6) for the groups $O(1)$ and $SO(2)$ [89]. When the matrix $C$ is not positive semidefinite, an approximation algorithm with ratio $\Omega(1/\log n)$ is given in [89] for the cases over $O(1)$ and $SO(2)$.

The generalization of the results in [89] to the cases over $O(d)$ and $SO(d)$ will be the subject of a separate publication. Overall, compared to the semidefinite relaxation in the generalized Procrustes analysis, our semidefinite relaxation has lower complexity yet enjoying similar theoretical guarantees.
2.1.2 Spectral Relaxations

Spectral relaxations for approximating the least squares solution have been previously considered in [84, 43, 112, 54, 14, 6, 98]. All methods are based on eigenvectors of the graph connection Laplacian (a notion that was introduced in [87]) or one of its normalizations. The graph connection Laplacian, denoted $L_1$ is constructed as follows: define the symmetric matrix $W_1 \in \mathbb{R}^{nd \times nd}$, such that the $(i,j)$’th $d \times d$ block is given by $(W_1)_{ij} = w_{ij} R_{ij}$. Also, let $D_1 \in \mathbb{R}^{nd \times nd}$, be a diagonal matrix such that $(D_1)_{ii} = d_i I_d$, where $d_i = \sum_j w_{ij}$. The graph connection Laplacian $L_1$ is defined as

$$L_1 = D_1 - W_1.$$

(2.8)

It can be verified that $L_1$ is PSD, and that in the noiseless case $L_1 R^T = 0$. The least $d$ eigenvectors of $L_1$ corresponding to its smallest eigenvalues, followed by the SVD procedure for rounding (2.7) can be used to recover the rotations. A slightly modified procedure that uses the eigenvectors of the normalized graph connection Laplacian

$$\mathcal{L}_1 = D_1^{-1/2} L_1 D_1^{-1/2} = I_{nd} - D_1^{-1/2} W_1 D_1^{-1/2}$$

(2.9)

is analyzed in [6]. Specifically, Theorem 10 in [6] bounds the least squares cost (2.1) incurred by this approximate solution from above and below in terms of the eigenvalues of the normalized graph connection Laplacian and the second eigenvalue of the normalized Laplacian (the latter reflecting the fact that synchronization is easier on well-connected graphs, or equivalently, more difficult on graphs with bottlenecks). This generalizes a previous result obtained by [96] that considers a spectral relaxation algorithm for MAX-CUT that achieves a non-trivial approximation ratio.
2.2 Least Unsquared Deviations (LUD) and Semidefinite Relaxation

As mentioned earlier, the least squares approach may not be optimal when a large proportion of the measurements are outliers [62, 92, 47]. To guard the orientation estimation from outliers, we replace the sum of squared residuals in (2.1) with the more robust sum of unsquared residuals

\[
\min_{R_1, \ldots, R_n \in SO(d)} \sum_{(i,j) \in \mathcal{E}} \left\| R_i^{-1} R_j - R_{ij} \right\|, \quad (2.10)
\]

to which we refer as LUD\(^4\). The self consistency error given in (2.10) mitigates the contribution from large residuals that may result from outliers. However, the problem (2.10) is non-convex and therefore extremely difficult to solve if one requires the matrices \( R_i \) to be rotations, that is, when adding the orthogonality and determinant constraints of \( SO(d) \) given in (1.1).

Notice that the cost function (2.10) can be rewritten using the Gram matrix \( G \) that was defined earlier in (2.3) for the SDP relaxation. Indeed, the optimization (2.10) is equivalent to

\[
\min_{R_1, \ldots, R_n \in SO(d)} \sum_{(i,j) \in \mathcal{E}} \| G_{ij} - R_{ij} \|. \quad (2.11)
\]

Relaxing the non-convex rank and determinant constraints as in the SDP relaxation leads to the following natural convex relaxation of the optimization problem (2.10):

\[
\min_{G} \sum_{(i,j) \in \mathcal{E}} \| G_{ij} - R_{ij} \| \quad \text{s.t.} \ G_{ii} = I_d, \ \text{and} \ G \succeq 0. \quad (2.12)
\]

\(^4\)For simplicity we consider the case where \( w_{ij} = 1 \) for \((i, j) \in \mathcal{E}\). In general, one may consider the minimization of \( \sum_{(i,j) \in \mathcal{E}} w_{ij} \| R_i^{-1} R_j - R_{ij} \| \).
This type of relaxation is often referred to as semidefinite relaxation (SDR) \[89, 50\]. Once \(G\) is found, either the deterministic or random procedures for rounding can be used to determine the rotations \(R_1, \ldots, R_n\).

### 2.3 Exact Recovery of the Gram matrix \(G\)

#### 2.3.1 Main Theorem

Consider the Gram matrix \(G\) as \(G = (G_{ij})_{i,j=1,...,n}\) where \(G_{ij} = R_i^T R_j\) is obtained by solving the minimization problem (2.12). We will show that for a certain probabilistic measurement model, the Gram matrix \(G\) is exactly recovered with high probability (w.h.p.). Specifically, in our model, the measurements \(R_{ij}\) are given by

\[
R_{ij} = \delta_{ij} R_i^T R_j + (1 - \delta_{ij}) \tilde{R}_{ij}
\]  

for \((i, j) \in \mathcal{E}\), where \(\delta_{ij}\) are i.i.d indicator Bernoulli random variables with probability \(p\) (i.e. \(\delta_{ij} = 1\) with probability \(p\) and \(\delta_{ij} = 0\) with probability \(1 - p\)). The incorrect measurements \(\tilde{R}_{ij}\) are i.i.d samples from the uniform (Haar) distribution over the group \(SO(d)\). Assume all pairwise ratios are measured, that is, \(\mathcal{E} = \{(i, j) | i, j = 1, \ldots, n\}\). Let \(\mathcal{E}_c\) denote the index set of correctly measured rotation ratios \(R_{ij}\), that is, \(\mathcal{E}_c = \{(i, j) | R_{ij} = R_i^T R_j\}\). In fact, \(\mathcal{E}_c\) is the edge set of a realization of a random graph drawn from the Erdős-Rényi model \(G(n, p)\). In the remainder of this section we shall prove the existence of a critical probability, denoted \(p^*_c(d)\), such that for all \(p > p^*_c(d)\) the program (2.12) recovers \(G\) from the measurements (2.13) w.h.p. (that tends to 1 as \(n \to \infty\)). In addition, we give an explicit upper bound \(p_c(d)\) for \(p^*_c(d)\).

---

\(^5\)The measurements in (2.13) satisfy \(R_{ij} = R_j R_i^T\), \(R_{ii} = I_d\).
Theorem 2.3.1 Assume that all pairwise ratio measurements $R_{ij}$ are generated according to (2.13). Then there exists a critical probability $p^*_c(d)$ such that when $p > p^*_c(d)$, the Gram matrix $G$ is exactly recovered by the solution to the optimization problem (2.12) w.h.p. (as $n \to \infty$). Moreover, an upper bound $p_c(d)$ for $p^*_c(d)$ is

$$p_c(d) = 1 - \left( \frac{-c_1(d) + \sqrt{c_1(d)^2 + 8 \left( c(d) + \frac{2}{\sqrt{d}} \right) / \sqrt{d}}}{2 \left( c(d) + \frac{2}{\sqrt{d}} \right)} \right)^2,$$  \hspace{0.5cm} (2.14)

where $c(d)$ is a constant defined as

$$c(d) = \frac{1}{d} \mathbb{E} \left( \text{Tr} \left( \frac{I_d - R}{\|I_d - R\|} \right) \right),$$  \hspace{0.5cm} (2.15)

$$c_1(d) = \sqrt{\frac{1 - c(d)^2}{2} d},$$  \hspace{0.5cm} (2.16)

where the expectation is w.r.t. the rotation $R$ that is distributed uniformly at random. In particular,

$$p_c(2) \approx 0.4570, \; p_c(3) \approx 0.4912, \; \text{and} \; p_c(4) \approx 0.5186$$

for $SO(2)$, $SO(3)$ and $SO(4)$ respectively.

Remark The case $d = 1$ is trivial since the special orthogonal group contains only one element in this case, but is not trivial for the orthogonal group $O(1)$. In fact, the latter is equivalent to MAX-CUT. Our proof below implies that $p_c(1) = 0$, that is, if the proportion of correct measurements is strictly greater than $1/2$, then all signs are recovered exactly w.h.p as $n \to \infty$. In fact, the proof below shows that w.h.p the

\hspace{2cm} \text{Remark 2.3.3: The value of } c(d) \text{ is discussed in section 2.3.3.}
signs are recovered exactly if the bias of the proportion of good measurements is as small as $O\left(\sqrt{\frac{\log n}{n}}\right)$.

### 2.3.2 Proof of Theorem 2.3.1

We shall show that the correct solution is a global minimum of the objective function, and analyze the perturbation of the objective function directly. The proof proceeds in two steps. First, we show that without loss of generality we can assume that the correct solution is $R_i = I_d$ and $G_{ij} = I_d$ for all $1 \leq i, j \leq n$. Then, we consider the projection of the perturbation into three different subspaces. Using the fact that the diagonal blocks of the perturbation matrix must be 0, it is possible to show that when the perturbation reduces the objective function for indices in $\mathcal{E}\setminus\mathcal{E}_c$ (that is, the incorrect measurements), it has to increase the objective function for indices in $\mathcal{E}_c$. If the success probability $p$ is large enough, then w.h.p. the amount of increase (to which we later refer as the “loss”) is greater than the amount of decrease (to which we later refer as the “gain”), therefore the correct solution must be the solution of the convex optimization problem (2.12).

**Fixing the correct solution**

**Lemma 2.3.2** If the optimal solution of (2.12) is w.h.p. $G_{ij} = I_d$ when $R_i = I_d$, then the optimal solution of (2.12) is w.h.p. $G_{ij} = R_i^T R_j$ for arbitrary rotations $\{R_i\}$.

**Proof** We give a bijection that preserves feasibility and objective value between the feasible solutions to (2.12) when $R_i = I_d$ and the solution for general $R_i$.

In fact, given any feasible solution $G$ for general $R_i$, let

$$\hat{G} = \text{diag} \left( R_1, \ldots, R_n \right) G \text{ diag} \left( R_1, \ldots, R_n \right)^T.$$ 

---

7The method of decomposing the perturbation was used in [47] and [113] for analyzing the performance of a convex relaxation method for robust principal component analysis.
that is, $\hat{G}_{ij} = R_i G_{ij} R_j^T$. If we rotate $R_{ij}$ similarly to get $\hat{R}_{ij} = R_i R_{ij} R_j^T$, then $\hat{R}_{i,j} = I$.

Since $G$ is the solution of (2.12), we know $\hat{G}$ must be a solution to the following convex program with the same objective value

$$\min_{\hat{G}} \sum_{(i,j) \in E} \| \hat{G}_{ij} - \hat{R}_{ij} \| \quad \text{s.t.} \quad \hat{G}_{ii} = I_d, \text{ and } \hat{G} \succeq 0. \quad (2.17)$$

However, observe that for edges in $E_c$, $R_i R_{ij} R_j^T = I_d$; for edges not in $E_c$, $R_i R_{ij} R_j^T$ is still a uniformly distributed random rotation in $SO(d)$ (due to left and right invariance of the Haar measure). Therefore, (2.17) is equivalent to (2.12) when $R_i = I_d$.

The other direction can be proved identically.

Using the Lemma 2.3.2, we can assume without loss of generality that $R_i = I_d$ for all $1 \leq i \leq n$. Now the correct solution should be $G_{ij} = I_d$. We denote this solution by $G$, and consider a perturbation $G + \Delta$.

**Decomposing the perturbation**

Let $G + \Delta$ be a perturbation of $G$ such that $\Delta_{ii} = 0$ and $G + \Delta \succeq 0$. Let $S$ be the $d$-dimensional linear subspace of $\mathbb{R}^{dn}$ spanned by the vectors $s_1, \ldots, s_d$, given by

$$s_1 = (e_1, e_1, \ldots, e_1)^T, \quad s_2 = (e_2, e_2, \ldots, e_2)^T, \quad \vdots \quad s_d = (e_d, e_d, \ldots, e_d)^T, \quad (2.18)$$

where $e_m (m = 1, 2, \ldots, d)$ is the $d$-dimensional row vector

$$e_m(l) = \begin{cases} 1 & \text{if } l = m, \\ 0 & \text{otherwise}, \end{cases}$$
that is, \( e_1 = (1, 0, \ldots, 0), e_2 = (0, 1, 0, \ldots, 0), \ldots \) and \( e_d = (0, \ldots, 0, 1) \).

Intuitively, if a vector \( v \) is in the space \( S \), then the restrictions of \( v \) on blocks of size \( d \) are all the same (i.e. \( v_{1, \ldots, d} = v_{d+1, \ldots, 2d} = \cdots = v_{(n-1)d+1, \ldots, nd} \)). On the other hand, if a vector \( v \) is in \( \bar{S} \), the orthogonal complement of \( S \), then it satisfies

\[
\sum_{i=1}^{n} v_{(i-1)d+1, \ldots, id} = 0.
\]

For two linear subspaces \( A, B \), we use \( A \otimes B \) to denote the space spanned by vectors \( \{ a \otimes b | a \in A \text{ and } b \in B \} \). By properties of tensor operation, the dimension of \( A \otimes B \) equals the dimension of \( A \) times the dimension of \( B \). We also identify the tensor product of two vectors \( a \otimes b \) with the (rank 1) matrix \( ab^T \).

Now let \( P, Q \) and \( T \) be \( \Delta \)'s projections to \( S \otimes S, (S \otimes \bar{S}) \cup (\bar{S} \otimes S), \) and \( \bar{S} \otimes \bar{S} \) respectively, that is,

\[
\Delta = P + Q + T, \quad \text{where } P \in S \otimes S, \; Q \in (S \otimes \bar{S}) \cup (\bar{S} \otimes S), \; \text{and } T \in \bar{S} \otimes \bar{S}.
\]

Using the definition of \( S \), it is easy to verify that

\[
P_{ij} = P_{11} \quad \forall 1 \leq i, j \leq n, \quad (2.19)
\]

\[
Q = Q^1 + Q^2, \quad (2.20)
\]

where \( Q^1 \in S \otimes \bar{S}, Q^2 \in \bar{S} \otimes S, Q^1 = (Q^2)^T, \)

\[
Q^1_{ij} = Q^1_{2j} = \cdots = Q^1_{nj}, \forall j \quad (2.21)
\]

and

\[
Q^2_{i1} = Q^2_{i2} = \cdots = Q^2_{in}, \forall i. \quad (2.22)
\]
Moreover, we have
\[
\sum_j Q_{1j}^1 = 0, \quad \sum_i Q_{i1}^2 = 0 \quad \text{and} \quad \sum_{i,j} T_{ij} = 0.
\]
(2.23)

For the matrix \( T \), the following notation is used to refer to its submatrices
\[
T_{ij} = (T_{ij}^{pq})_{p,q=1,\ldots,d}, \quad \text{and} \quad T^{pq} = (T_{ij}^{pq})_{i,j=1,\ldots,n},
\]
where \( T_{ij} \) are \( T \)'s \( d \times d \) sub-blocks and \( T^{pq} \) are \( T \)'s \( n \times n \) submatrices whose entries are the \((p,q)\)'th elements of the sub-blocks \( T_{ij} \).

Recall that \( G_{ij} = I_d \). Denote the objective function by \( F \), that is,
\[
F(G) = \sum_{(i,j) \in E} \|G_{ij} - R_{ij}\|.
\]
(2.24)

Then,
\[
F(G + \Delta) - F(G) = \sum_{(i,j) \in E} (\|I_d - R_{ij} + \Delta_{ij}\| - \|I_d - R_{ij}\|)
= \sum_{(i,j) \in E \setminus E_c} (\|I_d - R_{ij} + \Delta_{ij}\| - \|I_d - R_{ij}\|) + \sum_{(i,j) \in E_c} \|\Delta_{ij}\|
=: f_1 + f_2.
\]
(2.25)

Intuitively, if the objective value of \( G + \Delta \) is smaller than the objective value of \( G \), then \( \Delta_{ij} \) should be close to 0 for the correct measurements \((i, j) \in E_c\), and is large on \((i, j) \) such that \((i, j) \notin E_c\).

We shall later show that the “gain” \( f_1 \) from \((i, j) \notin E_c\) can be upperbounded by the trace of \( T \) and the off-diagonal entries of \( Q_{ij}^1 \). Then we shall show that when the trace of \( T \) is large, the diagonal entries of \( \Delta_{ij} \) for \((i, j) \in E_c\) are large, therefore the “gain” is smaller than the “loss” generated by these diagonal entries. On the other hand, when the off-diagonal entries of \( Q_{ij}^1 \) are large, then the off-diagonal entries of
Δ_{ij} for (i, j) ∈ EC are large, once again the “gain” will be smaller than the “loss” generated by the off-diagonal entries.

**Observations on P, Q and T**

To bound the “gain” and “loss”, we need the following set of observations.

**Lemma 2.3.3** \(nP_{11} = -\sum_{i=1}^{n} T_{ii}\).

**Proof** Using (2.19)-(2.23) and the fact that \(\Delta_{ii} = 0\), we have

\[
0 = \sum_{i} \Delta_{ii} = \sum_{i} (P_{ii} + Q_{1i}^1 + Q_{2i}^2 + T_{ii})
= nP_{11} + \sum_{i} Q_{1i}^1 + \sum_{i} Q_{2i}^2 + \sum_{i} T_{ii} = nP_{11} + \sum_{i} T_{ii}.
\]

**Lemma 2.3.4** \((P_{ij} + Q_{ij}) + (P_{ji} + Q_{ji}) = -(T_{ii} + T_{jj})\).

**Proof** We use the symmetry of the matrices P, Q:

\[
(P_{ij} + Q_{ij}) + (P_{ji} + Q_{ji})
= P_{ii} + Q_{ij}^1 + Q_{ij}^2 + P_{jj} + Q_{ji}^1 + Q_{ji}^2
= P_{ii} + Q_{jj}^1 + Q_{jj}^2 + P_{jj} + Q_{ii}^1 + Q_{ii}^2
= (P_{ii} + Q_{ii}^1 + Q_{ii}^2) + (P_{jj} + Q_{jj}^1 + Q_{jj}^2)
= (\Delta_{ii} - T_{ii}) + (\Delta_{jj} - T_{jj})
= -(T_{ii} + T_{jj}),
\]

where the second equality follows (2.21)-(2.22).

**Lemma 2.3.5** \(T \succ 0\).

**Proof** Since \(G + \Delta = G + P + Q + T \succ 0\), for any vector \(v \in \tilde{S}\), \(v^T(G + \Delta)v \geq 0\). However, \(v^T Gv = v^T P v = v^T Q v = 0\) according to the definition of G, P and Q.
Therefore, for all $v \in \bar{S}$ we have $v^T T v \geq 0$. Also, $T w = 0$ for $w \in S$. Therefore, if $v \in \bar{S}$ and $w \in S$ then $(v + w)^T T (v + w) = v^T T v \geq 0$. Hence, $T$ is positive semidefinite.

**Lemma 2.3.6** Let $T_i^d$ be a diagonal matrix whose diagonal entries are those of $T_{ij}$, then $\| \sum_i T_i^d \| \geq \text{Tr}(T) / \sqrt{d}$.

**Proof** This is just a straightforward application of the Cauchy-Schwarz inequality.

Denote $X = \sum_i T_i^d$. Clearly, $X \succeq 0$, since it is a sum of positive semidefinite matrices. Let $x_1, \ldots, x_d$ be the diagonal entries of $X$. Then, from Cauchy-Schwarz inequality, we have

$$\text{Tr}(X) = \sum_j x_j \leq \left( \sum_j x_j^2 \right)^{\frac{1}{2}} \cdot \sqrt{d} = \sqrt{d} \| X \|,$$

that is, $\| \sum_i T_i^d \| = \| X \| \geq \text{Tr}(X) / \sqrt{d} = \text{Tr}(T) / \sqrt{d}$.

**Lemma 2.3.7** Let $A$ be a $n \times n$ adjacency matrix such that $A_{ij} = 1$ if and only if $(i, j) \in E_c$; otherwise, $A_{ij} = 0$. Let $B = A - \frac{1}{n^2} (1^T A 1) 11^T$, where 1 is the all-ones (column) vector. Let $\lambda = \| B \|_2$, where $\| \cdot \|_2$ denotes the spectral norm of a matrix. Then,

$$\lambda = O_P (\sqrt{n}).$$

Here the notation $f = O_P (g)$ means $f$ is upper bounded by $c_p \cdot g$ with high probability, where $c_p$ is a constant that may depend on $p$.

**Proof** Let $B_1 = A - p 11^T$. Observe that $B_1$ is a random matrix where each off-diagonal entry is either $(1 - p)$ with probability $p$ or $-p$ with probability $(1 - p)$. Therefore, by Wigner’s semi-circle law and the concentration of the eigenvalues of random symmetric matrices with i.i.d entries of absolute value at most 1 [1], the
largest eigenvalue (in absolute value) of $B_1$ is $\|B_1\|_2 = O_P(\sqrt{n})$. Then we have

$$\|B\|_2 = \left\| B_1 + \left( p - \frac{1}{n^2} (1^T A 1) \right) 11^T \right\|_2$$

$$\leq \|B_1\|_2 + \left\| p - \frac{1}{n^2} (1^T A 1) \right\| 11^T$$

$$\leq \|B_1\|_2 + n O_P \left( n^{-1} \sqrt{\log n} \right)$$

$$= O_P(\sqrt{n}),$$

where the second inequality uses the Chernoff bound.

**Remark** The matrix $A$ is the adjacency matrix of the Erdős–Rényi (ER) random graph model $G(n, p)$ where the expectation of every node’s degree is $(n - 1)p$. Denote by $\lambda_i$ the $i$’th largest eigenvalue of a matrix. It is intuitive to see that $\lambda_1(A) \approx (n - 1)p$ and the first eigenvector is approximately the all-ones vector $1$, and $\lambda_2(A) \approx \lambda_1(B) = O_P(\sqrt{n})$.

Using these observations, we can bound the sum of norm of all $T_{ij}$’s by the trace of $T$.

**Lemma 2.3.8** We have the following three inequalities for the matrix $T$:

1. $\left| \sum_{(i,j) \in E_c} T_{ij}^{pq} \right| \leq \lambda Tr(T^{pp})^{\frac{1}{2}} Tr(T^{qq})^{\frac{1}{2}}$ for $p, q = 1, 2, \ldots, d$,

2. $\left\| I_d, \sum_{(i,j) \in E_c} T_{ij} \right\| \leq \lambda Tr(T)$,

3. $\left\| \sum_{(i,j) \in E_c} T_{ij} \right\| \leq \lambda Tr(T)$.

Here the notation for the inner product of two matrices $\langle X, Y \rangle$ means $\text{Tr}(X^TY)$.

**Proof** 1. Since $T \succ 0$, we can assume $T_{ij}^{pq} = \langle u_i^p, u_j^q \rangle$, where $u_i^p(i = 1, \ldots, n, p = 1, \ldots, d)$ is an $n$ dimensional row vector, then $\text{Tr}(T^{pp}) = \sum_{i=1}^n \langle u_i^p, u_i^p \rangle$, and

$$\sum_{(i,j) \in E_c} T_{ij}^{pq} = (u_1^p, \ldots, u_n^p) (A \otimes I_n) (u_1^q, \ldots, u_n^q)^T.$$
We claim that $\sum_{i=1}^{n} u_i^p = 0$. In fact, since $T \in \bar{S} \otimes \bar{S}$, for any $p, q = 1, \ldots, d$, we have $\sum_{i,j=1}^{n} T_{ij}^{pq} = s_p^T T s_q = 0$. Therefore, we obtain

$$0 = \sum_{i,j=1}^{n} T_{ij}^{pp} = \sum_{i,j=1}^{n} \langle u_i^p, u_j^p \rangle = \left( \sum_{i=1}^{n} u_i^p, \sum_{j=1}^{n} u_j^p \right) = \left\| \sum_{i=1}^{n} u_i^p \right\|^2$$

and thus we have

$$\sum_{i=1}^{n} u_i^p = 0 \text{ for } p = 1, \ldots, d. \quad (2.26)$$

Let $v_m^p$ $(m = 1, \ldots, n, p = 1, \ldots, d)$ be a $n$ dimensional row vector such that

$$v_m^p = (u_1^p(m), u_2^p(m), \ldots, u_n^p(m)),$$

then we have $\sum_{m=1}^{n} \left\| v_m^p \right\|^2 = \text{Tr} (T^{pp})$, and $\langle v_m^p, 1 \rangle = 0$ due to (2.26). Therefore

$$\left| \sum_{(i,j) \in E_c} T_{ij}^{pq} \right| = \left| \left( u_1^p, \ldots, u_n^p \right) (A \otimes I_n) \left( u_1^q, \ldots, u_n^q \right)^T \right|$$

$$= \left| \sum_{m=1}^{n} v_m^p A (v_m^q)^T \right|$$

$$\leq \sum_{m} \lambda \left\| v_m^p \right\| \left\| v_m^q \right\|$$

$$\leq \lambda \left( \sum_{m} \left\| v_m^p \right\|^2 \right)^{\frac{1}{2}} \left( \sum_{m} \left\| v_m^q \right\|^2 \right)^{\frac{1}{2}},$$

$$= \lambda \text{Tr} (T^{pp})^{\frac{1}{2}} \text{Tr} (T^{qq})^{\frac{1}{2}}$$

where the first inequality uses Lemma 2.3.7 and the fact that $v_m^p$ $(m = 1, \ldots, n, p = 1, \ldots, d)$ is orthogonal to the all-ones vector $1$, and the second inequality uses Cauchy-Schwarz inequality.

2. $\left| \left\langle I_d, \sum_{(i,j) \in E_c} T_{ij} \right\rangle \right| \leq \lambda \text{Tr} (T)$ is clear from 1. when $p = q$.  

32
3. From 1. we have

\[
\left\| \sum_{(i,j) \in \mathcal{E}_c} T_{ij} \right\| = \sqrt{\sum_{p,q} \left( \sum_{(i,j) \in \mathcal{E}_c} T_{ij}^{pq} \right)^2} \
\leq \sqrt{\sum_{p,q} \left( \lambda \text{Tr} (T_{pp})^{\frac{1}{2}} \text{Tr} (T_{qq})^{\frac{1}{2}} \right)^2} \
= \lambda \sqrt{\sum_{p,q} \text{Tr} (T_{pp}) \text{Tr} (T_{qq})} \
= \lambda \sum_{p} \text{Tr} (T_{pp}) \
= \lambda \text{Tr} (T) .
\]

Bounding the “gain” from incorrect measurements

To make the intuition that “gain” from incorrect measurements is always upper bounded by the “loss” from correct measurements formal, we shall first focus on \((i, j) \not\in \mathcal{E}_c\), and bound the “gain” by trace of \(T\) and norms of \(Q_{ij}^1\).

Recall that \(f_1\) in (2.25) is the “gain”, we shall bound it using the following lemma.

**Lemma 2.3.9** For any pair of non-zero matrices \(M_1, M_2 \in \mathbb{R}^{m \times m}\), we have

\[
\|M_1 + M_2\| - \|M_1\| \geq \langle M_1 / \|M_1\| , M_2 \rangle . \tag{2.27}
\]

**Proof** Using Cauchy-Schwarz inequality, we obtain

\[
\|M_1\|^2 + \langle M_1 , M_2 \rangle = \langle M_1 , M_1 + M_2 \rangle \leq \|M_1\| \|M_1 + M_2\| ,
\]

which is equivalent to (2.27).
We apply Lemma 2.3.9 to the “gain” $f_1$ and obtain

$$f_1 \geq \sum_{(i,j) \in \mathcal{E}\setminus \mathcal{E}_c} \left\langle \frac{I_d - R_{ij}}{\|I_d - R_{ij}\|}, \Delta_{ij} \right\rangle$$

$$= \sum_{(i,j) \in \mathcal{E}\setminus \mathcal{E}_c} \left\langle \frac{I_d - R_{ij}}{\|I_d - R_{ij}\|}, P_{ij} + Q_{ij} + T_{ij} \right\rangle .$$  \hspace{1cm} (2.28)

First we shall bound the “gain” from the matrix $P$. Since blocks of $P$ are the same, the average

$$\frac{1}{|\mathcal{E}\setminus \mathcal{E}_c|} \sum_{(i,j) \in \mathcal{E}\setminus \mathcal{E}_c} \left\langle \frac{I_d - R_{ij}}{\|I_d - R_{ij}\|}, P_{ij} \right\rangle$$

should be concentrated around the expectation of $\left\langle \frac{I_d - R_{ij}}{\|I_d - R_{ij}\|}, P_{ij} \right\rangle$. The expectation is analyzed in section 2.3.3. By (2.19)-(2.22) and the law of large numbers, we obtain that

$$\sum_{(i,j) \in \mathcal{E}\setminus \mathcal{E}_c} \left\langle \frac{I_d - R_{ij}}{\|I_d - R_{ij}\|}, P_{ij} \right\rangle$$

$$= \left( c(d) (1 - p) n (n - 1) + O_P \left( n \sqrt{\log n} \right) \right) \text{Tr} (P_{11})$$

$$= - \left( c(d) (1 - p) (n - 1) + O_P \left( \sqrt{\log n} \right) \right) \text{Tr} (T),$$  \hspace{1cm} (2.29)

where the last equality uses Lemma 2.3.3. $c(d)$ is defined in (2.15) and the rotation matrix $R$ is uniformly sampled from the group $SO(d)$ (see section 2.3.3 for detailed discussion).

For matrix $Q$ we use similar concentration bounds.
\[
\sum_{(i,j) \in \mathcal{E} \setminus \mathcal{E}_c} \left\langle \frac{I_d - R_{ij}}{\|I_d - R_{ij}\|}, Q_{ij} \right\rangle = \sum_{(i,j) \in \mathcal{E} \setminus \mathcal{E}_c} \left\langle \frac{I_d - R_{ij}}{\|I_d - R_{ij}\|}, Q_{ij}^1 + Q_{ij}^2 \right\rangle \\
= \sum_{(i,j) \in \mathcal{E} \setminus \mathcal{E}_c} \left\langle \frac{I_d - R_{ij}}{\|I_d - R_{ij}\|}, Q_{ij}^1 + Q_{ij}^2 \right\rangle \\
\geq \sum_{j} (n - p(n - 1)) \langle c(d) I_d, Q_{jj}^1 \rangle + \sum_{i} (n - p(n - 1)) \langle c(d) I_d, Q_{ii}^2 \rangle \\
- O_P \left( \sqrt{n \log n} \right) \sum_{i} \|Q_{jj}^1\| - O_P \left( \sqrt{n \log n} \right) \sum_{i} \|Q_{ii}^2\| \\
= (n - p(n - 1)) \left\langle cI_d, \sum_{j} Q_{jj}^1 \right\rangle + (n - p(n - 1)) \left\langle cI_d, \sum_{i} Q_{ii}^2 \right\rangle \\
- O_P \left( \sqrt{n \log n} \right) \sum_{i} \|Q_{ii}^1\| \\
= - O_P \left( \sqrt{n \log n} \right) \sum_{i} \|Q_{ii}^1\|, \tag{2.30}
\]

where the third equality uses the fact that \(Q^1 = (Q^2)^T\), and the last equality follows (2.23).

Finally we shall bound the “gain” from matrix \(T\), which is

\[
\sum_{(i,j) \in \mathcal{E} \setminus \mathcal{E}_c} \left\langle \frac{I_d - R_{ij}}{\|I_d - R_{ij}\|}, T_{ij} \right\rangle. \tag{2.31}
\]

Before continuing, we need the following results in Lemma 2.3.10 for a matrix \(D\), where \(D\) is defined as

\[
D_{ij} = \begin{cases} 
\frac{I_d - R_{ij}}{\|I_d - R_{ij}\|} - c(d) I_d & \text{for } (i,j) \in \mathcal{E} \setminus \mathcal{E}_c, \\
0 & \text{otherwise.} 
\end{cases} \tag{2.32}
\]

The proof of Lemma 2.3.10 is in section 2.3.4.
Lemma 2.3.10  The limiting spectral density of \( \frac{1}{\sqrt{n-1}} D \) is Wigner’s semi-circle. In addition, the matrix \( D \) can be decomposed as \( D = D_+ + D_- \), where \( D_+ \succ 0 \) and \( D_- \prec 0 \), and we have

\[
\|D_+\|^2 \approx \|D_-\|^2 \approx \frac{1}{2} (1 - p) n (n - 1) \left( 1 - c(d)^2 d \right).
\]  

(2.33)

We return to lower bounding \((2.31)\). Since \( \sum_{ij} T_{ij} = 0 \), we have

\[
\sum_{(i,j) \in \mathcal{E} \setminus \mathcal{E}_c} \left\langle \frac{I_d - R_{ij}}{\|I_d - R_{ij}\|}, T_{ij}\right\rangle
\]

\[
= \sum_{(i,j) \in \mathcal{E} \setminus \mathcal{E}_c} \langle c(d) I_d, T_{ij}\rangle + \sum_{(i,j) \in \mathcal{E} \setminus \mathcal{E}_c} \left\langle \frac{I_d - R_{ij}}{\|I_d - R_{ij}\|}, -c(d) I_d, T_{ij}\right\rangle
\]

\[
= -c(d) \sum_{(i,j) \in \mathcal{E}_c} \langle I_d, T_{ij}\rangle - c(d) \sum_i \langle I_d, T_{ii}\rangle + \sum_{ij} \langle D_{ij}, T_{ij}\rangle
\]

\[
= -c(d) \sum_{(i,j) \in \mathcal{E}_c} \langle I_d, T_{ij}\rangle - c(d) \text{Tr}(T) + \text{Tr}(DT)
\]

\[
= -c(d) \sum_{(i,j) \in \mathcal{E}_c} \langle I_d, T_{ij}\rangle - c(d) \text{Tr}(T) + \text{Tr}(D_+ T) + \text{Tr}(D_- T)
\]

\[
\geq -c(d) \sum_{(i,j) \in \mathcal{E}_c} \langle I_d, T_{ij}\rangle - c(d) \text{Tr}(T) + \text{Tr}(D_- T), \quad (2.34)
\]

where the last inequality follows from the fact that \( \text{Tr}(D_+ T) \geq 0 \) since both \( T \) and \( D_+ \) are positive semidefinite matrices. Recall that for any positive semidefinite matrix \( X \) the following inequality holds: \( \|X\| \leq \text{Tr}(X) \). Since \( T \succ 0 \), using \((2.33)\) we obtain

\[
|\text{Tr}(D_- T)| \leq \|D_-\| \|T\| \leq \|D_-\| \text{Tr}(T) \approx \frac{1}{\sqrt{2}} \sqrt{(1 - p) \sqrt{1 - c(d)^2 d} \text{Tr}(T)}. \quad (2.35)
\]

Also, Lemma 2.3.8 reads

\[
\left| \left\langle I_d, \sum_{(i,j) \in \mathcal{E}_c} T_{ij}\right\rangle\right| \leq \lambda \text{Tr}(T). \quad (2.36)
\]
Combining (2.34), (2.35), and (2.36) together gives
\[
\sum_{(i,j) \in E \setminus E_c} \left\langle \frac{I_d - R_{ij}}{\|I_d - R_{ij}\|}, T_{ij} \right\rangle \geq - \left( c(d) \lambda + c(d) + \frac{1}{\sqrt{2}} \sqrt{(1-p)} \sqrt{1 - c(d)^2} \right) \text{Tr} (T). \tag{2.37}
\]
Since from Lemma 2.3.7 we have \( \lambda = O_P (\sqrt{n}) \), the bound is given by
\[
\sum_{(i,j) \in E \setminus E_c} \left\langle \frac{I_d - R_{ij}}{\|I_d - R_{ij}\|}, T_{ij} \right\rangle \geq - \left( \frac{1}{\sqrt{2}} \sqrt{(1-p)} \sqrt{1 - c(d)^2} \right) dn + O_P (\sqrt{n}) \text{Tr} (T). \tag{2.38}
\]
Combining (2.28), (2.29), (2.30) and (2.38) together we obtain a lower bound for the gain from \( \Delta \) as following:
\[
f_1 \geq - \left( \frac{1}{\sqrt{2}} \sqrt{(1-p)} \sqrt{1 - c(d)^2} dn + c(d) (1-p) n + O_P (\sqrt{n}) \right) \text{Tr} (T) - O_P (\sqrt{n} \log n) \sum_i \|Q_{ii}^\dagger\|. \tag{2.39}
\]

Bounding the “loss” from correct measurements

Now we consider the part \( f_2 = \sum_{(i,j) \in E_c} \|\Delta_{ij}\| \), which is the loss from the correct entries. We use the notations \( \Delta_{ij}^d \) and \( \Delta_{ij}^{off} \) to represent the restrictions of the submatrices \( \Delta_{ij} \) on the diagonal entries and off-diagonal entries respectively. We will analyze \( \sum_{(i,j) \in E_c} \|\Delta_{ij}^d\| \) and \( \sum_{(i,j) \in E_c} \|\Delta_{ij}^{off}\| \) separately.

For the diagonal entries we have
\[
\sum_{(i,j) \in \mathcal{E}_c} \| \Delta_{ij}^d \| = \sum_{(i,j) \in \mathcal{E}_c} \| P_{ij}^d + Q_{ij}^d + T_{ij}^d \|
\]
\[
\geq \left\| \sum_{(i,j) \in \mathcal{E}_c} (P_{ij}^d + Q_{ij}^d + T_{ij}^d) \right\|
\]
\[
\geq \left\| \sum_{(i,j) \in \mathcal{E}_c} P_{ij}^d + Q_{ij}^d \right\| - \left\| \sum_{(i,j) \in \mathcal{E}_c} T_{ij}^d \right\|
\]
\[
= \left\| \sum_{(i,j) \in \mathcal{E}_c} T_{ii}^d + T_{jj}^d \right\| - \left\| \sum_{(i,j) \in \mathcal{E}_c} T_{ij}^d \right\|
\]
\[
= pn \left\| \sum_i T_{ii}^d \right\| - \mathcal{O}_P \left( \sqrt{n \log n} \right) \text{Tr} (T) - \left\| \sum_{(i,j) \in \mathcal{E}_c} T_{ij}^d \right\|
\]
\[
\geq \frac{2pn}{\sqrt{d}} \text{Tr} (T) - \mathcal{O}_P \left( \sqrt{n \log n} \right) \text{Tr} (T) - \lambda \text{Tr} (T)
\]
\[
= \left( \frac{pn}{\sqrt{d}} - \mathcal{O}_P \left( \sqrt{n \log n} \right) \right) \text{Tr} (T), \quad (2.40)
\]

where the second equality uses Lemma 2.3.4, the third equality uses the law of large numbers and Chernoff bound, the last inequality follows Lemma 2.3.6 and Lemma 2.3.8, and the last equality uses the fact that \( \lambda = \mathcal{O}_P(\sqrt{n}) \) from Lemma 2.3.7.

For the off-diagonal entries, we have the following lemma, whose proof is deferred to section 2.3.5.

**Lemma 2.3.11**

\[
\sum_{(i,j) \in \mathcal{E}_c} \| \Delta_{ij}^{off} \| \geq \frac{p}{d^2} \left( \mathcal{O}_P (n) \sum_{i=1}^n \| Q_{ii}^1 \| - \mathcal{O}_P (n) \text{Tr} (T) \right). \quad (2.41)
\]

Finally, we can merge the loss in two parts by a simple inequality:
Lemma 2.3.12 If $\sum_i |a_i| \geq a$ and $\sum_i |b_i| \geq b$, then for any $\alpha, \beta > 0$ such that $\alpha^2 + \beta^2 = 1$, we have

$$\sum_i \sqrt{a_i^2 + b_i^2} \geq \alpha a + \beta b.$$ 

Applying Lemma 2.3.12 to (2.40) and (2.41), and setting $\alpha = 1 - \frac{1}{\sqrt{n}}$ and $\beta = \sqrt{\frac{2}{\sqrt{n}} - \frac{1}{n}}$, we obtain a lower bound for the loss from $\Delta$ when $p > \frac{1}{2}$:

$$f_2 = \sum_{(i,j) \in E_c} \|\Delta_{ij}\|$$

$$\geq \left(1 - \frac{1}{\sqrt{n}}\right) \sum_{(i,j) \in E_c} \|\Delta_{ij}^d\| + \sqrt{\frac{2}{\sqrt{n}} - \frac{1}{n}} \sum_{(i,j) \in E_c} \|\Delta_{ij}^{off}\|$$

$$= \left(\frac{2pn}{\sqrt{d}} - \mathcal{O}_P\left(n^{3/4}\right)\right) \text{Tr} (T) + \mathcal{O}_P\left(n^{3/4}\right) \sum_{i=1}^n \|Q_{ii}^1\|. \quad (2.42)$$

Finishing the proof

Now that we have bounded the "gain" and the "loss", we just need a condition on $p$ such that the "loss" is greater than the "gain" (w.h.p.). Combining (2.25), (2.39) and (2.42) together, when $p > 1/2$ we obtain

$$F(G + \Delta) - F(G) = f_1 + f_2$$

$$\geq \left(\frac{2p}{\sqrt{d}} - \frac{1}{\sqrt{2}} \sqrt{(1-p)\sqrt{1 - c(d)^2} - c(d)(1-p)}\right) n - \mathcal{O}_P\left(n^{3/4}\right) \text{Tr} (T)$$

that is, when the number of rotations $n$ is large enough, we just need

$$p \gtrsim 1 - \left(-c_1(d) + \sqrt{c_1(d)^2 + 8\left(c(d) + 2/\sqrt{d}\right) / \sqrt{d}} \right)^2 2\left(c(d) + 2/\sqrt{d}\right) := p_c(d),$$

where $c_1(d)$ is defined as (2.16)
### 2.3.3 The Value of $c(d)$

Now let us study the constant $c(d)$ such that $c(d)I_d = \mathbb{E}\left(\frac{I_d - R}{\|I_d - R\|}\right)$, where $R$ is uniformly sampled from the rotation group $SO(d)$. Due to the symmetry of the uniform distribution over the rotation group $SO(d)$, the off-diagonal entries of $\mathbb{E}\left(\frac{I_d - R}{\|I_d - R\|}\right)$ are all zeros, and the diagonal entries of $\mathbb{E}\left(\frac{I_d - R}{\|I_d - R\|}\right)$ are all the same as the constant $c(d)$.

Using the fact that

$$\|I_d - R\| = \sqrt{\text{Tr}\left((I_d - R)(I_d - R)^T\right)} = \sqrt{\text{Tr}(2I_d - R - R^T)} = \sqrt{2\text{Tr}(I_d - R)},$$

we have

$$c(d) = \frac{1}{d} \mathbb{E}\left(\frac{\text{Tr}(I_d - R)}{\|I_d - R\|}\right) = \frac{1}{d} \mathbb{E}\left(\frac{\text{Tr}(I_d - R)}{\sqrt{2\text{Tr}(I_d - R)}}\right) = \frac{1}{\sqrt{2d}} \mathbb{E}\left(\sqrt{\text{Tr}(I_d - R)}\right),$$

where the probability measure for expectation is the Haar measure. The function $f(R) = \sqrt{\text{Tr}(I_d - R)}$ is a class function, that is, $f(R)$ is invariant under conjugation, meaning that for all $O, R \in SO(d)$ we have $f(OR^T) = f(R)$. Therefore $\mathbb{E}(f(R))$ can be computed using the Weyl integration formula specialized to $SO(d)$ (Exercise 40).
\[ \int_{SO(2m+1)} f(R) \, dR = \frac{1}{2^m m!} \int_{[-\pi,\pi]^m} f \left( \begin{pmatrix} \cos \theta_1 & -\sin \theta_1 \\ \sin \theta_1 & \cos \theta_1 \\ & \ddots \\ & & \cos \theta_m & -\sin \theta_m \\ & & \sin \theta_m & \cos \theta_m \\ & & & \cdots \\ & & & & 1 \end{pmatrix} \right) \prod_{i<j} \left| e^{i\theta_i} - e^{i\theta_j} \right|^2 \left| e^{i\theta_i} - e^{-i\theta_j} \right|^2 \prod_{i} \left| e^{i\theta_i} - 1 \right|^2 \, d\theta_1 \ldots d\theta_m, \] (2.44)

\[ \int_{SO(2m)} f(R) \, dR = \frac{1}{2^{m-1} m!} \int_{[-\pi,\pi]^m} f \left( \begin{pmatrix} \cos \theta_1 & -\sin \theta_1 \\ \sin \theta_1 & \cos \theta_1 \\ & \ddots \\ & & \cos \theta_m & -\sin \theta_m \\ & & \sin \theta_m & \cos \theta_m \\ & & & \cdots \\ & & & & 1 \end{pmatrix} \right) \prod_{i<j} \left| e^{i\theta_i} - e^{i\theta_j} \right|^2 \left| e^{-i\theta_i} - e^{-i\theta_j} \right|^2 \, d\theta_1 \ldots d\theta_m. \] (2.45)
In particular, for $d = 2$ or $3$ ($m = 1$), using the Weyl integration formula we obtain

\[
c(2) = \frac{1}{2\sqrt{2}} \mathbb{E} \left( \sqrt{\text{Tr} (I_2 - R)} \right)
\]

\[
= \frac{1}{2\sqrt{2}} \cdot \frac{1}{2\pi} \int_{-\pi}^{\pi} \text{Tr} \left( I_2 - \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \right) d\theta
\]

\[
= \frac{1}{4\pi} \int_{-\pi}^{\pi} \sqrt{1 - \cos \theta} d\theta
\]

\[
= \frac{1}{2\pi} \int_{0}^{\pi} \sqrt{1 - \cos \theta} d\theta \quad \text{(let } x = \cos \theta)\]

\[
= \frac{1}{2\pi} \int_{-1}^{1} \frac{1}{\sqrt{1 + x}} dx
\]

\[
= \sqrt{2}/\pi,
\]

and

\[
c(3) = \frac{1}{3\sqrt{2}} \mathbb{E} \left( \sqrt{\text{Tr} (I_3 - R)} \right)
\]

\[
= \frac{1}{3\sqrt{2}} \cdot \frac{1}{2\pi} \int_{-\pi}^{\pi} \text{Tr} \left( I_3 - \begin{pmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \right) (1 - \cos \theta) d\theta
\]

\[
= \frac{1}{3\pi} \int_{-\pi}^{\pi} \sqrt{1 - \cos \theta} (1 - \cos \theta) d\theta
\]

\[
= \frac{2\sqrt{2}}{3\pi} \int_{0}^{\pi} \sin \frac{\theta}{2} \left( 1 - \cos^2 \frac{\theta}{2} \right) d\theta \quad \text{(let } x = \cos \frac{\theta}{2})
\]

\[
= \frac{4\sqrt{2}}{3\pi} \int_{0}^{1} (1 - x^2) dx
\]

\[
= \frac{8\sqrt{2}}{9\pi}.
\]

For $d \geq 4$, the computation of $c(d)$ involves complicated multiple integrals. Now we study the lower and upper bound and the limiting value of $c(d)$ for large $d$.

**Lemma 2.3.13** \( \frac{1}{2\sqrt{2[d/2]}} \leq c(d) \leq \frac{1}{\sqrt{2d}} \), where \( \lfloor \cdot \rfloor \) denotes the floor of a number.
\textbf{Proof} Using the fact that the square root function is concave, we obtain

\begin{align*}
c(d) &= \frac{1}{\sqrt{2d}} \mathbb{E} \left( \sqrt{\text{Tr} (I_d - R)} \right) \\
&\leq \frac{1}{\sqrt{2d}} \mathbb{E} (\text{Tr} (I_d - R)) \\
&= \frac{\sqrt{d}}{\sqrt{2d}} = \frac{1}{\sqrt{2d}}
\end{align*}

where the second equality uses the fact that \( \text{Tr} (R) = 0 \) due to the symmetry of the Haar measure.

Since \( d - 4 \lfloor d/2 \rfloor \leq \text{Tr} (R) \leq d \), we have

\begin{equation}
0 \leq \text{Tr} (I_d - R) \leq 4 \lfloor d/2 \rfloor . \tag{2.46}
\end{equation}

In the range \((2.46)\), \( \sqrt{\text{Tr} (I_d - R)} \geq \frac{1}{2\sqrt{\lfloor d/2 \rfloor}} \text{Tr} (I_d - R) \) due to the concavity of the square root function. Therefore we obtain

\begin{align*}
c(d) &= \frac{1}{\sqrt{2d}} \mathbb{E} \left( \sqrt{\text{Tr} (I_d - R)} \right) \\
&\geq \frac{1}{2d \sqrt{\lfloor d/2 \rfloor}} \mathbb{E} (\text{Tr} (I_d - R)) \\
&= \frac{d}{2d \sqrt{\lfloor d/2 \rfloor}} = \frac{1}{2 \sqrt{\lfloor d/2 \rfloor}}
\end{align*}

Remark. The upper bound of \( c(d) \) is very close to \( c(d) \) for \( d = 2, 3, \) and 4:

\begin{align*}
\frac{1}{2\sqrt{2}} (\approx 0.3536) &\leq c(2) = \frac{\sqrt{2}}{\pi} (\approx 0.4502) \leq \frac{1}{2}, \\
\frac{1}{2\sqrt{2}} (\approx 0.3536) &\leq c(3) = \frac{8\sqrt{2}}{9\pi} (\approx 0.4001) \leq \left( \frac{1}{\sqrt{6}} \approx 0.4082 \right), \\
0.25 = \frac{1}{4} &\leq c(4) (\approx 0.3505) \leq \frac{1}{2\sqrt{2}} (\approx 0.3536).
\end{align*}

In fact we can prove the following lemma.
Lemma 2.3.14  \( \lim_{d \to \infty} \sqrt{2d} c(d) = 1. \)

Proof  We have

\[
1 \geq \sqrt{2d} \ c(d) = \frac{1}{\sqrt{d}} \mathbb{E} \left( \sqrt{\text{Tr} \ (I_d - R)} \right) \\
\geq \frac{1}{\sqrt{d}} \mathbb{P} \left( |\text{Tr} \ (R)| \leq d^{\frac{1}{2}} \right) \sqrt{d - d^{\frac{1}{2}}}, \tag{2.47}
\]

where the first inequality follows Lemma 2.3.13. Diaconis and Mallows [19] showed that the moments of the trace of \( R \) equal the moments of a standard normal variable for all sufficiently large \( n \). In particular, when \( d \to \infty \), the limit of \( \text{Tr} \ (R) \) has mean 0 and variance 1. Therefore using Chebyshev inequality we get that

\[
\mathbb{P} \left( |\text{Tr} \ (R)| \leq d^{\frac{1}{2}} \right) \to 1 \text{ as } d \to \infty. \tag{2.48}
\]

Hence let \( d \to \infty \) in (2.47), we obtain \( \lim_{d \to \infty} \sqrt{2d} c(d) = 1. \)

2.3.4 Proof of Lemma 2.3.10

Here we aim to prove that the limiting spectral density of the normalized random matrix \( \frac{1}{\sqrt{n}} D \) is Wigner’s semi-circle, where \( D \) is defined in (2.32). The following definitions and results will be used.

- The Cauchy (Stieltjes) transform of a probability measure \( \mu \) and moments \( \{m_k\}_{k=1}^{\infty} \) is given by

\[
G_{\mu}(z) = \int_{\mathbb{R}} \frac{d\mu(x)}{z - x} = \frac{1}{z} + \sum_{k=1}^{\infty} \frac{m_k}{z^{k+1}}, \ z \in \mathbb{C}, \ \text{Im}(z) > 0.
\]

- The density \( g(x) \) can be recovered from \( G_{\mu} \) by the Stieltjes inversion formula:

\[
g(x) = -\frac{1}{\pi} \lim_{\epsilon \to 0} \text{Im} \left( G_{\mu}(x + i\epsilon) \right). \]
• The Wigner semicircle distribution \( \mu_{0, \sigma^2} \) centered at 0 with variance \( \sigma^2 \) is the distribution with density

\[
g(x) = \begin{cases} \frac{\sqrt{4\sigma^2 - x^2}}{2\pi\sigma^2} & \text{if } x \in [-2\sigma, 2\sigma], \\ 0 & \text{otherwise}. \end{cases}
\]

• The Cauchy transform of the semicircle distribution \( \mu_{0, \sigma^2} \) is given by

\[
G_{\mu_{0, \sigma^2}}(z) = \frac{z - \sqrt{z^2 - 4\sigma^2}}{2\sigma^2}.
\]

We now state a theorem by Girko, which extends the semicircle law to the case of random block matrices, and show how, in particular, this follows for the random matrix \( \frac{1}{\sqrt{n}} D \).

**Theorem 2.3.15** (Girko, 1993 in [29]) Suppose the \( nd \times nd \) matrix \( M \) is composed of \( d \times d \) independent random blocks \( M_{ij}, i, j = 1, 2, \ldots, n \) which satisfy

\[
M_{ij} = M_{ji}^T, \quad E(M_{ij}) = O_d, \quad E\|M_{ij}\|^2 < \infty. \tag{2.49}
\]

Suppose also that

\[
\sup_n \max_{i=1,2,\ldots,n} \sum_{j=1}^n E\|M_{ij}\|^2 < \infty \tag{2.50}
\]

and that Lindeberg’s condition holds: for every \( \epsilon > 0 \)

\[
\lim_{n \to \infty} \max_{i=1,2,\ldots,n} \sum_{j=1}^n E \left( \|M_{ij}\|^2 \mathcal{X}(\|M_{ij}\| > \epsilon) \right) = 0, \tag{2.51}
\]

where \( \mathcal{X} \) denotes an indicator function. Let \( \lambda_1 \geq \ldots \geq \lambda_{nd} \) be the eigenvalues of \( M \) and let \( \mu_n(x) = \frac{1}{nd} \sum_{l=1}^{nd} \mathcal{X}(\lambda_l < x) \) be the eigenvalue measure. Then for almost any \( x \),

\[
|\mu_n(x) - F_n(x)| \to 0 \quad \text{as } n \to \infty \quad \text{a.s.,}
\]

45
where \( F_n(x) \) are the distribution functions whose Cauchy/Stieltjes transforms are given by \( \frac{1}{nd} \sum_{i=1}^{n} \text{Tr}(C_i(z)) \), where the matrices \( C_i(z) \) satisfy

\[
C_i(z) = \left( \mathbb{E} \left( zI_d - \mathbb{E} \sum_{j=1}^{n} M_{ij}C_j(z) M_{ij}^T \right) \right)^{-1}, \quad i = 1, 2, \ldots, n. \tag{2.52}
\]

The solution \( C_i(z), i = 1, 2, \ldots, n \) to (2.52) exists and is unique in the class of analytic \( d \times d \) matrix functions \( C(z) \) for which \( \text{Im}(z) \text{Im}(C(z)) < 0, \text{Im}(z) \neq 0 \).

Now we apply the result of the theorem on the matrix \( W = \frac{1}{\sqrt{n-1}} D \) whose \( d \times d \) blocks satisfy

\[
W_{ij} = \begin{cases} 
\frac{1}{\sqrt{n-1}} \left( \frac{I_d - R_{ij}}{\|I_d - R_{ij}\|} - c(d) I_d \right) & \text{w.p. } 1-p, \\
0 & \text{w.p. } p,
\end{cases}
\]

for \( i \neq j \), where \( R_{ij} = R_{ji}^T \). Notice that \( W \) is a random symmetric matrix \( (W_{ij} = W_{ji}^T) \) with i.i.d off-diagonal blocks. From the definition of \( c(d) \) it follows that \( \mathbb{E}(W_{ij}) = 0 \). Also, \( \|W_{ij}\| \) is finitely bounded, that is,

\[
\|W_{ij}\|^2 = \frac{1}{n-1} \left( 1 + c(d)^2 d - \frac{c(d)}{\sqrt{2}} \sqrt{\text{Tr}(I_d - R_{ij})} \right) \leq \frac{1}{n-1} \left( 1 + c(d)^2 d \right).
\]

Therefore, all the conditions (condition (2.49)-(2.51)) of the theorem are satisfied by the matrix \( W \). Before we continue, we need the following

**Lemma 2.3.16** We have \( \mathbb{E} \left( \frac{I_d - R}{\|I_d - R\|^2} \right) = \frac{1}{2d} I_d \), and \( \mathbb{E} \left( \frac{I_d - R}{\|I_d - R\|^2} \right) = c(d) I_d \).

**Proof** Using the symmetry of the Haar measure on \( SO(d) \), we can assume that \( \mathbb{E} \left( \frac{I_d - R}{\|I_d - R\|^2} \right) = aI_d \), and \( \mathbb{E} \left( \frac{I_d - R}{\|I_d - R\|^2} \right) = bI_d \), where \( a \) and \( b \) are constants depending on
Therefore we have

\[
a = \frac{1}{d} \mathbb{E} \left( \text{Tr} \left( \frac{I_d - R}{\|I_d - R\|^2} \right) \right) = \frac{1}{d} \mathbb{E} \left( \frac{\text{Tr} (I_d - R)}{\text{Tr} (I_d - R) (I_d - R)^T} \right) = \frac{1}{d} \mathbb{E} \left( \frac{\text{Tr} (I_d - R)}{2 \text{Tr} (I_d - R)} \right) = \frac{1}{2d},
\]

and

\[
b = \frac{1}{d} \mathbb{E} \left( \text{Tr} \left( \frac{I_d - R}{\|I_d - R\|^2} \right) \right) = c(d).
\]

We claim that \( C_i(z) = h(z) I_d \), where \( h(z) \) is a function of \( z \). In fact,

\[
\sum_{j=1}^{n} W_{ij} C_j(z) W_{ij}^T = h(z) \sum_{j=1}^{n} W_{ij} W_{ij}^T = (1 - p) h(z) \mathbb{E} \left( \frac{I_d - R}{\|I_d - R\|^2} - c(d) I_d \right) \left( \frac{I_d - R}{\|I_d - R\|^2} - c(d) I_d \right)^T
\]

\[
= (1 - p) h(z) \mathbb{E} \left( 2 \frac{I_d - R - R^T}{\|I_d - R\|^2} - 2c(d) \frac{I_d - R}{\|I_d - R\|} + c(d)^2 I_d \right)
\]

\[
= (1 - p) h(z) \left( \frac{1}{d} I_d - 2c(d)^2 I_d + c(d)^2 I_d \right)
\]

\[
= (1 - p) h(z) \left( \frac{1}{d} - c(d)^2 \right) I_d,
\]

where the fourth equality uses Lemma 2.3.16. From (2.52) we obtain

\[
h(z) I_d = \left( z I_d - (1 - p) h(z) \left( \frac{1}{d} - c(d)^2 \right) I_d \right)^{-1},
\]

47
that is, 

\[
(1 - p) \left( \frac{1}{d} - c(d)^2 \right) h(z)^2 - zh(z) + 1 = 1,
\]

which reduces to 

\[
h(z) = \frac{z \pm \sqrt{z^2 - 4 (1 - p) \left( \frac{1}{d} - c(d)^2 \right)}}{2 (1 - p) \left( \frac{1}{d} - c(d)^2 \right)}.
\]

The uniqueness of the solution now implies that the Cauchy transform of \( F_n(x) \) is 

\[
\frac{1}{nd} \sum_{i=1}^{n} \text{Tr} (C_i(z)) = h(z) = \frac{z - \sqrt{z^2 - 4 (1 - p) \left( \frac{1}{d} - c(d)^2 \right)}}{2 (1 - p) \left( \frac{1}{d} - c(d)^2 \right)},
\]

where we select the \(-\) branch according to the properties of the Cauchy transform. This is exactly the Cauchy transform of the semicircle law with support 

\[
\left[ -2 \sqrt{(1 - p) \left( \frac{1}{d} - c(d)^2 \right) (n-1)}, 2 \sqrt{(1 - p) \left( \frac{1}{d} - c(d)^2 \right) (n-1)} \right].
\]

Hence, for large \( n \) the eigenvalues of \( D \) are distributed according to the semicircle law with support 

\[
\left[ -2 \sqrt{(1 - p) \left( \frac{1}{d} - c(d)^2 \right) (n-1)}, 2 \sqrt{(1 - p) \left( \frac{1}{d} - c(d)^2 \right) (n-1)} \right].
\]

Thus the limiting spectral density of \( \frac{1}{\sqrt{n-1}} D \) is Wigner’s semi-circle. Since any symmetric matrix can be decomposed into a superposition of a positive semidefinite matrix and a negative definite matrix (this follows immediately from the spectral decomposition of the matrix), the matrix \( D \) can be decomposed as 

\[
D = D_+ + D_-,
\]

where \( D_+ \succ 0 \) and \( D_- \prec 0 \). Clearly, 

\[
\|D\|^2 = \|D_+\|^2 + \|D_-\|^2,
\]

48
and since the limiting spectral density of $\frac{1}{\sqrt{n-1}}D$ is Wigner’s semi-circle that is symmetric around 0, we have

$$\|D_+\|^2 \approx \|D_-\|^2 \approx \frac{1}{2} \|D\|^2.$$  

From the law of large numbers we get that $\|D\|^2$ is concentrated at

$$\|D\|^2 \approx (1 - p) n (n - 1) \mathbb{E} \left( \left\| \frac{I_d - R_{ij}}{\|I_d - R_{ij}\|} - c(d) I_d \right\|^2 \right)$$

$$= (1 - p) n (n - 1) \left( 1 + c(d)^2 d - 2c(d) \mathbb{E} \left( \text{Tr} \left( \frac{I_d - R}{\|I_d - R\|} \right) \right) \right)$$

$$= (1 - p) n (n - 1) \left( 1 - c(d)^2 d \right).$$

Hence,

$$\|D_+\|^2 \approx \|D_-\|^2 \approx \frac{1}{2} (1 - p) n (n - 1) \left( 1 - c(d)^2 d \right).$$

2.3.5 Proof of Lemma 2.3.11

In (2.30), we have already shown that the gain from $Q$ is at most $\mathcal{O}_P(\sqrt{n \log n}) \sum_{i=1}^n \|Q_{ii}^1\|$. Now we will show that the gain from $Q$ is less than the sum of $\mathcal{O}_P(\sqrt{n \log n}) \text{Tr} (T)$ and the loss in the off-diagonal entries of $\Delta_{ij}$ for $(i, j) \in \mathcal{E}_c$, specifically speaking,

$$\mathcal{O}_P \left( \sqrt{n \log n} \sum_{i=1}^n \|Q_{ii}^1\| \right) \leq \mathcal{O}_P \left( \sqrt{n \log n} \text{Tr} (T) + \frac{d^2}{pn} \mathcal{O}_P \left( \sqrt{n \log n} \right) \sum_{(i,j) \in \mathcal{E}_c} \|\Delta_{ij}^{\text{off}}\|, \right. \tag{2.53}$$

which is equivalent to prove Lemma 2.3.11.

A matrix is skew-symmetric if $X^T = -X$. Every matrix $X$ can be written as the sum of a symmetric matrix and a skew-symmetric matrix: $X = \frac{X + X^T}{2} + \frac{X - X^T}{2}$. We will call the first term the symmetric part and the second term the skew-symmetric
part. Recall that $Q^{1}$ is a matrix with identical columns ($Q^{1}_{ji} = Q^{1}_{ii}$). We denote the symmetric part and the skew-symmetric part of $Q^{1}_{ji}$ as

$$Q^{1,s}_{i} = Q^{1}_{ii} + Q^{2}_{ii} \frac{T_{ii} + P_{ii}}{2}$$

(2.54)

and

$$Q^{1,ss}_{i} = Q^{1}_{ii} - Q^{2}_{ii} \frac{2}{2}$$

(2.55)

respectively. Then we have

$$\|Q^{1}_{ii}\| = \|Q^{1,s}_{i} + Q^{1,ss}_{i}\| \leq \|Q^{1,s}_{i}\| + \|Q^{1,ss}_{i}\| .$$

(2.56)

Later we will see the skew-symmetric part $Q^{1,ss}_{i}$ is exactly what’s causing troubles. Before dealing with the trouble, let’s first handle the symmetric part $Q^{1,s}_{i}$.

For the symmetric part $Q^{1,s}_{i}$ we have

$$\sum_{i=1}^{n} \|Q^{1,s}_{i}\| \leq \frac{1}{2} \sum_{i=1}^{n} \|T_{ii} + P_{ii}\|$$

$$\leq \frac{1}{2} \left( \sum_{i=1}^{n} \|T_{ii}\| + \sum_{i=1}^{n} \|P_{ii}\| \right)$$

$$= \frac{1}{2} \left( \sum_{i=1}^{n} \|T_{ii}\| + \sum_{i=1}^{n} \|T_{ii}\| \right)$$

$$\leq \sum_{i=1}^{n} \|T_{ii}\| \leq \sum_{i=1}^{n} \text{Tr} (T_{ii}) = \text{Tr} (T) ,$$

(2.57)

where the second equality uses Lemma 2.3.3 and the third inequality uses the fact that $T_{ii} \succeq 0$ and that for any matrix $X \succeq 0$, $\|X\| \leq \text{Tr} (X)$.

Let us consider now the skew-symmetric part $Q^{1,ss}_{i}$. For any index $p,q \in \{1, 2, ..., d\}$, consider $\sum_{i=1}^{n} |Q^{1,ss}_{i} (p,q)|$, where $Q^{1,ss}_{i} (p,q)$ is the $(p,q)$th entry of the matrix $Q^{1,ss}_{i}$. Without loss of generality and for the purpose of simplicity, let us assume that $\sum_{i} |Q^{1,ss}_{i} (p,q)|$ is largest when $p = 1$ and $q = 2$ (it won’t be largest on
diagonal, because diagonal entries have to be 0). Now we know
\[ \sum_{i=1}^{n} \|Q_{1,ss}^i\| \leq d^2 \sum_{i=1}^{n} |Q_{1,ss}^i(1,2)|. \] (2.58)

This is just because $Q_{1,ss}^i$ has $d^2$ entries. Thus later we will just care about $Q_{1,ss}^i(1,2)$
and $Q_{1,ss}^i(2,1)$. We define the matrices $Q^{1,s}$, $Q^{1,ss}$, $Q^{2,s}$, and $Q^{2,ss}$ as
\[ Q_{ij}^{1,s} = Q_{ji}^{1,s}, \quad Q_{ij}^{1,ss} = Q_{ji}^{1,ss}, \quad Q_{ij}^{2,s} = Q_{ji}^{1,s}, \quad Q_{ij}^{2,ss} = -Q_{ji}^{1,ss} \text{ for all } i \text{ and } j, \] (2.59)

and the matrix $Q^s = Q^{1,s} + Q^{2,s}$ and $Q^{ss} = Q^{1,ss} + Q^{2,ss}$. Thus it is easy to see
$Q = Q^s + Q^{ss}$. Restrict the matrices $P$, $Q$, $T$ and so on to the good entries and
obtain $P_c$, $Q_c$, $T_c$ and so on (which means, for example, when $(i, j) \notin \mathcal{E}_c$, set the
$(i, j)$’s block $P_c(i, j) = 0$; when $(i, j) \in \mathcal{E}_c$, keep $P_c(i, j) = P(i, j)$. Then we can
prove the following lemmas.

**Lemma 2.3.17** Let the vectors $s_1, s_2 \in \mathbb{R}^{nd}$ be defined as in (2.18), that is,
\[ s_1 = (1, 0, 0, \ldots, 0, 1, 0, 0, \ldots, 0, \ldots, \ldots, 1, 0, 0, \ldots, 0)^T, \]
\[ s_2 = (0, 1, 0, \ldots, 0, 0, 1, 0, \ldots, 0, \ldots, \ldots, 0, 1, 0, \ldots, 0)^T. \]

And define the vectors $a_1, a_2 \in \mathbb{R}^{nd}$ as following:
\[ a_l(m) = \begin{cases} 1 & \text{if } m = d \cdot i + l \text{ and } Q^{1,ss}_i(p, q) > 0, \\ -1 & \text{if } m = d \cdot i + l \text{ and } Q^{1,ss}_i(p, q) < 0, \\ 0 & \text{otherwise,} \end{cases} \quad \text{for } l = 1, 2. \] (2.60)
Then we have the following inequality

\[ s_1^T (P_c + Q_c) a_2 - s_2^T (P_c + Q_c) a_1 \geq \left( \Omega(pn) - O_P \left( \sqrt{n \log n} \right) \right) \sum_{i=1}^{n} |Q_{i}^{1,ss} (1, 2)|. \]

(2.61)

**Proof** Firstly, note that for any matrix \( X \in \mathbb{R}^{nd \times nd} \), \( s_1^T X a_2 - s_2^T X a_1 \) is the sum of the differences between \( X_{ij} (1, 2) \) and \( X_{ij} (2, 1) \). Thus it is easy to verify that the symmetric matrix \( P_c \) and the symmetric parts \( Q_{1}^{1,ss} \) and \( Q_{2}^{2,ss} \) in \( Q_{ss}^{c} \) contribute 0 to the LHS of (2.61), that is,

\[ s_1^T (P_c + Q_c) a_2 - s_2^T (P_c + Q_c) a_1 = s_1^T Q_{ss}^{c} a_2 - s_2^T Q_{ss}^{c} a_1 = 2s_1^T Q_{ss}^{c} a_2, \]

(2.62)

where the second equality uses the fact that \( -s_2^T Q_{ss}^{c} a_1 = s_1^T Q_{ss}^{c} a_2 \) due to the skew-symmetry of the submatrices of \( Q_{ss}^{c} \).

---

Figure 2.1: Two \( n \times n \) matrices. The matrix \( M_a \) is defined as \( M_a(i, j) = |Q_{i}^{1,ss} (1, 2)| \) if \( (i, j) \in \mathcal{E}_c \), otherwise \( M_a(i, j) = 0 \). The matrix \( M_b \) is defined as \( M_b(i, j) = -Q_{i}^{1,ss} (1, 2) \text{ sign} \left( Q_{j}^{1,ss} (1, 2) \right) \) if \( (i, j) \in \mathcal{E}_c \), otherwise \( M_b(i, j) = 0 \). With the assumption that there is an integer \( i_0 \) such that \( Q_{i}^{1,ss} \geq 0 \) when \( i \leq i_0 \) and \( Q_{i}^{1,ss} < 0 \) when \( i > i_0 \), \( M_a \) and \( M_b \) have non-negative regions with sign “+” and non-positive regions with sign “-” as shown in sub-figure (a) and (b). The RHS of (2.63) and (2.64) can be represented as sums of the entries \( (i, j) \in \mathcal{E}_c \) of \( M_a \) and \( M_b \) respectively. When taking a summation of the RHS of (2.63) and (2.64), the entries in the “-” regions of \( M_b \) and those corresponding entries in \( M_a \) cancel each other out. Thus the contribution to the sum comes from the entries at \( (i, j) \in \mathcal{E}_c \) in “+” regions of \( M_b \) and the corresponding parts in \( M_a \).
Now we just need to focus on $s_1^T Q_{ss}^1 a_2 = s_1^T (Q_{ss}^1 + Q_{ss}^2) a_2$. For each of the submatrices of $Q_{ss}^1$ and $Q_{ss}^2$, we only look at their entry at index $(1, 2)$. Due to the definition in (2.59) and (2.60) we have

$$s_1^T Q_{ss}^1 a_2 = \sum_{(i,j) \in E_c} Q_{ij}^{ss} (1, 2) \text{ sign} (Q_{ij}^{ss} (1, 2)) = \sum_{(i,j) \in E_c} |Q_{ij}^{ss} (1, 2)| \quad (2.63)$$

and

$$s_1^T Q_{ss}^2 a_2 = \sum_{(i,j) \in E_c} Q_{ij}^{ss} (1, 2) \text{ sign} (Q_{ij}^{ss} (1, 2)) = \sum_{(i,j) \in E_c} -Q_{ij}^{ss} (1, 2) \text{ sign} (Q_{ij}^{ss} (1, 2)). \quad (2.64)$$

Without loss of generality and for the purpose of simplicity, we assume there is an integer $i_0$ such that $Q_{i}^{ss} \geq 0$ when $i \leq i_0$ and $Q_{i}^{ss} < 0$ when $i > i_0$. Then the RHS of (2.63) and (2.64) can be represented as sums of the entries $(i, j) \in E_c$ of the $n \times n$ matrices in Figure 2.1a and Figure 2.1b respectively. Apparently in some parts, even if there is a edge in $E_c$ things will cancel (these are the parts that are marked negative in Figure 2.1b). But in the parts that are marked positive in Figure 2.1b, if there is an edge in $E_c$, things will not cancel and will indeed contribute to $s_1^T Q_{ss}^c a_2$. In fact, the contribution of “+” regions is at least

$$(\Omega(pn) - O_P \left( \sqrt{n \log n} \right)) \sum_{i=1}^n |Q_i^{ss} (1, 2)| \leq 2s_1^T Q_{ss}^c a_2 \quad (2.65)$$

where we use Lemma 2.3.19 and the fact that

$$\sum_{Q_i^{ss} (1, 2) > 0} Q_i^{ss} (1, 2) = - \sum_{Q_i^{ss} (1, 2) < 0} Q_i^{ss} (1, 2) = \sum_{i=1}^n |Q_i^{ss} (1, 2)| / 2.$$
To verify that the contribution of "+" regions is lower bounded by (2.65), we first assume that in Figure 2.1, every good edge in the "+" area of (b) contributes the same amount (that corresponds to \( Q_{1,ss}^i \)) to the same absolute value for all \( i \), then there are two cases:

1. The rectangle has large area, in this case we will show ALL large rectangles have a large number of good edges.

2. The rectangle has small area, in this case the "+" area must be very wide (in order to have small area, it must be a \( q \times (n - q) \) rectangle where \( q \) is very small), in fact its width \((n - q)\) will be much wider than \( n/2 \), and will even be wider than \((1 - p)n\) (where \( p \) is the probability of a good edge), so there must be many good edges inside the rectangle.

**Lemma 2.3.18** Let \( p \) be the probability of a good edge. When \( p > \log C n \) for some fixed constant \( C \), with high probability for any set \( S_1, S_2 \subset \{1, 2, ..., n\} \), where \( |S_1| = q \) and \( |S_2| \geq n - C_2 q \) (\( C_2 = 16 \) should be good enough), the number of good edges in \( S_1 \times S_2 \) is at least \( C_3 p |S_1||S_2| \) (\( C_3 \) is some universal constant, and can be \( 1/4 \)).

**Proof** With high probability every vertex has at least \( pn/2 \) good edges, so when \( C_2 q < pn/4 \) the Lemma holds trivially.

When \( C_2 q \geq pn/4 \), use Chernoff bound. Fix the size of \( S_1 \) and \( S_2 \), for any particular \( S_1, S_2 \), the probability that the number of good edges is too small is bounded by

\[
e^{-\Omega((1-C_3)p|S_1||S_2|/\sqrt{p|S_1||S_2|})^2} = e^{-\Omega(pqn)}
\]

On the other hand, the number of such pairs \( S_1, S_2 \) is at most \( n^q \cdot n^{C_2 q} = e^{O(q \log n)} \), by union bound the probability that there exists a bad pair is very small (because \( q \log n \ll pqn \)).
Now we are essentially done if all the positive entries of $Q_{1,ss}^1(1,2)$ are of similar size. We are not really done because of the following bad case: One of $Q_{1,ss}^1(1,2)$ is very large, the others are small. Now although there are many good edges in the “+” region, their real contribution depends on how many of the good edges belong to the vertex with large $Q_{1,ss}^1(1,2)$. Thus we need to group positive entries of $Q_{1,ss}^1(1,2)$ according to their value, and apply Lemma 2.3.18 to different groups.

Lemma 2.3.19

\[ s^T_1 Q^*_c a_2 = s^T_1 Q^1_{ss} a_2 + s^T_1 Q^{2,ss}_c a_2 \geq (\Omega(pn) - \mathcal{O}_P(\sqrt{n \log n})) \sum_{Q_{1,ss}^1(1,2) < 0} -Q_{1,ss}^1(1,2) \]  

(2.66)

Proof Assume without loss of generality that the number of negative entries in $Q_{1,ss}^1(1,2)$ is less than $n/2$ (otherwise take $-Q$, the loss will be the same).

Let $N$ be the set of nonnegative entries of $Q_{1,ss}^1(1,2)$. Let $m$ be the minimum entry in $Q_{1,ss}^1(1,2)$ (which is negative). Let $S_j (j \geq 0)$ be the set of entries that are between $[m \cdot 2^{-j}, m \cdot 2^{-j-1})$.

We shall consider the rectangles $S_j \times (\{1, 2, ..., n\} - S_{j-1} - S_j - S_{j+1})$, we would like to show that the number of good edges in all these rectangles are at least $C_3 p |S_j| \cdot n/2$ (the second set is larger than $n/2$ because it contains $N$), and each of these good edges will contribute at least $-m \cdot 2^{-j-2}$ (remember $m$ is negative). In fact, these edges must be from $S_j$ to something in $S_{j-2}$ or $S_{j+2}$ or even farther index, the difference between any entry in $S_j$ and any entry in those sets are at least $2^{-j-2}$, thus they will not cancel completely. Therefore the total contribution will be at least

\[ \sum_{j \geq 0} C_3 p |S_j| \cdot n/2 \cdot (-m) \cdot 2^{-j-2} \geq (\Omega(pn) - \mathcal{O}_P(\sqrt{n \log n})) \sum_{Q_{1,ss}^1(1,2) < 0} -Q_{1,ss}^1(1,2). \]
We can show there are many good edges by Lemma 2.3.18, the only thing we need to guarantee is the second set \( \{1, 2, \ldots, n\} - S_{j-1} - S_j - S_{j+1} \) is large enough.

When \( \{1, 2, \ldots, n\} - S_{j-1} - S_j - S_{j+1} \) is small, there are two possibilities here:

1. \(|S_{j-1}| \) might be larger than \( C_2|S_j|/2 \), but in this case the sum of entries in \( S_{j-1} \) is constant times larger than \( S_j \), thus we can ignore \( S_j \).

2. \(|S_{j+1}| \) might be larger than \( C_2|S_j|/2 \), but we have chosen \( C_2 \) to be large enough, so in this case the sum of entries in \( S_{j+1} \) is also constant times larger than \( S_j \), thus we can ignore \( S_j \).

To clarify points 1 and 2, let \( h(j) \) be the absolute value of sum of entries in \( S_j \), i.e.,

\[
h(j) = \sum_{Q_i^{1, ss}(1, 2) < 0, i \in S_j} -Q_i^{1, ss}(1, 2).
\]  

(2.67)

We will show the contributions from the rectangles are at least

\[
(\Omega(pn) - O_P(\sqrt{n \log n}) \sum_j h(j).
\]

Points 1 and 2 say if \( h(j) \) is at most \( h(j+1)/2 \) or \( h(j-1)/2 \) then we ignore \( h(j) \). This is fine because if we look at a \( h(j) \) that is not ignored (there must be such sets, otherwise the sum of \( h(j) \) will be infinity), it can only be responsible for \( h(j-1), h(j-2), \ldots \) and \( h(j+1), h(j+2), \ldots \). And we know \( h(j - t) \leq 2^{-t}h(j) \) and \( h(j + t) \leq 2^{-t}h(j) \) (because they are all ignored). The sum of all these \( h(j - t)'s \) and \( h(j + t)'s \) are bounded by constant times \( h(j) \).

**Lemma 2.3.20**

\[
\sum_{i=1}^{n} \|Q_i^{1, ss}\| \leq \frac{d^2}{\Omega(pn) - O_P(\sqrt{n \log n})} \left( \sum_{(i,j) \in \mathcal{E}_c} \|\Delta_{ij}^{off}\| + \frac{n}{2} Tr(T) \right).
\]  

(2.68)
Proof We know that
\[ s_1^T (P_c + Q_c + T_c) a_2 - s_2^T (P_c + Q_c + T_c) a_1 = s_1^T \Delta_s a_2 - s_2^T \Delta_s a_1 \leq \sum_{(i,j) \in \mathcal{E}_c} |\Delta_{ij} (1,2)| + \sum_{(i,j) \in \mathcal{E}_c} |\Delta_{ij} (2,1)| \leq 2 \sum_{(i,j) \in \mathcal{E}_c} \| \Delta_{ij}^{\text{off}} \|, \] (2.69)

where $\Delta_{ij}^{\text{off}}$ is obtained by restricting $\Delta_{ij}$ on the off-diagonal entries. In addition, we have
\[ -s_1^T T_c a_2 \leq \sum_{i,j} |T_{ij} (1,2)| \leq \frac{1}{2} \sum_{i,j} (T_{ii} (1,1) + T_{jj} (2,2)) = \frac{n}{2} \text{Tr} (T), \]

where the second inequality uses the fact that $T \succeq 0$, and similarly we have $s_2^T T_c a_1 \leq \frac{n}{2} \text{Tr} (T)$, thus we obtain
\[ -s_1^T T_c a_2 + s_2^T T_c a_1 \leq n \text{Tr} (T). \] (2.70)

Combining (2.58), (2.61), (2.69) and (2.70) together we get (2.68).

Using (2.56), (2.57) and (2.68), we reach the conclusion in (2.53).

2.4 Stability of LUD

In this section we will analyze the behavior of LUD when the measurements $R_{ij}$ on the “good” edge set $\mathcal{E}_c$ are no longer the true rotation ratios $R_i^T R_j$, instead, $R_{ij}$ are small perturbations of $R_i^T R_j$. Similar to the noise model (2.13), we assume in our
model that the measurements $R_{ij}$ are given by

$$R_{ij} = \delta_{ij} \tilde{R}_{ij} + (1 - \delta_{ij}) \bar{R}_{ij},$$

(2.71)

where the rotation $\bar{R}_{ij}$ is sampled from a probability distribution (e.g. the von Mises-Fisher distribution \[13\]; c.f. section 2.7.1) such that

$$\mathbb{E} \left( \| \bar{R}_{ij} - R_i^T R_j \| \right) = \epsilon, \quad \text{and} \quad \text{Var} \left( \| \bar{R}_{ij} - R_i^T R_j \| \right) = \mathcal{O} \left( \epsilon^2 \right).$$

(2.72)

Note that the stability result is not limited to the random noise, and the analysis can also be applied to bounded deterministic perturbations.

We can generalize the analysis for exact recovery to this new noise model (2.71) with small perturbations on the “good” edge set and prove the following theorems for weak stability and strong stability of LUD.

### 2.4.1 Weak Stability

**Theorem 2.4.1 (Weak stability)** Assume that all pairwise ratio measurements $R_{ij}$ are generated according to (2.71) such that the condition (2.72) holds for a fixed small $\epsilon > 0$. Then there exists a critical probability $p^*_c(d)$ such that when $p > p^*_c(d)$, the solution $\hat{G}$ to the optimization problem (2.12) is close to the true Gram matrix $G$ in the sense that

$$\left\| G - \hat{G} \right\|^2 \leq \mathcal{O} \left( n^2 \right) \epsilon^2$$

w.h.p. (as $n \to \infty$). Moreover, an upper bound $p_c(d)$ for $p^*_c(d)$ is given by (2.14).

**Proof** First, the “gain” $f_1$ from the incorrect measurements remains the same, since the noise model for the “bad” edge set $\mathcal{E}\setminus\mathcal{E}_c$ is not changed. Thus, the lower bound
for $f_1$ is given in (2.39). For the “loss” from the good measurements, we have

$$f_2 = \sum_{(i,j) \in E_c} \left( \| I_d - R_{ij} + \Delta_{ij} \| - \| I_d - R_{ij} \| \right) \geq \sum_{(i,j) \in E_c} \left( \| \Delta_{ij} \| - 2 \| I_d - R_{ij} \| \right) \geq \sum_{(i,j) \in E_c} \| \Delta_{ij} \| - 2p \left( n^2 + \mathcal{O}_P \left( n \sqrt{\log n} \right) \right) \epsilon.$$ \hspace{1cm} (2.73)

Applying Lemma 2.3.12 to (2.40) and (2.41), and setting $\alpha = \sqrt{1 - \epsilon_0}$ and $\beta = \sqrt{\epsilon_0}$, where $\epsilon_0 \ll p - p_c$, we obtain

$$\sum_{(i,j) \in E_c} \| \Delta_{ij} \| \geq \sqrt{1 - \epsilon_0} \sum_{(i,j) \in E_c} \| \Delta_{ij}^d \| + \sqrt{\epsilon_0} \sum_{(i,j) \in E_c} \| \Delta_{ij}^{off} \| \geq c_2 n \text{Tr} (T) + c_3 n \sum_{i=1}^n \| Q_{1ii} \| - \mathcal{O}_P \left( \sqrt{n \log n} \right) \text{Tr} (T), \hspace{1cm} (2.74)$$

where $c_2$ and $c_3$ are some constants. Thus, combining (2.39), (2.73) and (2.74) together, we get

$$F(G + \Delta) - F(G) = f_1 + f_2 \geq c_4 n \text{Tr} (T) + c_3 n \sum_{i=1}^n \| Q_{1ii} \| - \mathcal{O}_P \left( \sqrt{n \log n} \right) \text{Tr} (T) - \mathcal{O}_P \left( \sqrt{n \log n} \right) \sum_{i=1}^n \| Q_{1ii} \| - 2p \left( n^2 + \mathcal{O}_P \left( n \sqrt{\log n} \right) \right) \epsilon, \hspace{1cm} (2.75)$$

where $c_4$ is some constant. Thus, if the RHS of (2.75) is greater than zero, then $G + \Delta$ is not the minimizer of $F$. In other words, if $G + \Delta$ is the minimizer of $F$, then the RHS of (2.75) is not greater than zero. Let $n \to \infty$ in (2.75), we obtain the necessary
condition for the minimizer $G + \Delta$ of $F$:

$$c_4 \text{Tr} (T) + c_3 \sum_{i=1}^{n} \|Q_{ii}\| \leq 2pn\epsilon. \quad (2.76)$$

To show that condition (2.76) leads to the conclusion that the amount of perturbation

$$\|\Delta\|^2 = \|P\|^2 + \|Q\|^2 + \|T\|^2. \quad (2.77)$$

is less than $O(n^2) \epsilon^2$, we need the following lemmas to upper bound $\|\Delta\|^2$ by parts.

**Lemma 2.4.2**

$$\|P\|^2 \leq \text{Tr} (T)^2 \leq O(n^2) \epsilon^2, \quad \text{and} \quad \|T\|^2 \leq \text{Tr} (T)^2 \leq O(n^2) \epsilon^2. \quad (2.78)$$

**Proof** Using the fact that for any positive semidefinite matrix $M$, $\|M\| \leq \text{Tr} (M)$, we have

$$\|P\|^2 = n^2 \|P_{11}\|^2 = \left\| \sum_{i=1}^{n} T_{ii} \right\|^2 \leq \text{Tr} \left( \sum_{i=1}^{n} T_{ii} \right)^2 = \text{Tr} (T)^2. \quad (2.79)$$

$$\|T\|^2 \leq \text{Tr} (T)^2. \quad (2.80)$$

And following (2.76) we obtain (2.78).

**Lemma 2.4.3**

$$\|Q\|^2 \leq 2 \text{Tr}(T)^2 - 2 \text{Tr}(T) + nd. \quad (2.81)$$

**Proof** Since $G + \Delta \succeq 0$, we can decompose $G + \Delta$ as $G + \Delta = \sum_{i=1}^{nd} \lambda_i v_i v_i^T$, where the eigenvectors $v_i \in \mathbb{R}^{nd}$ and the associated eigenvalues $\lambda_i \geq 0$ for all $i$. Then we further decompose each $v_i$ as $v_i = v_i^s + v_i^\bar{s}$, where $v_i^s \in \mathcal{S}$ and $v_i^\bar{s} \in \bar{\mathcal{S}}$. Thus we
obtain

\[ G + \Delta = \sum_{i=1}^{nd} \lambda_i v_i v_i^T = \sum_{i=1}^{nd} \lambda_i \left( v_i^S + v_i^S \right) \left( v_i^S + v_i^S \right)^T \]

(2.82)

\[ = \sum_{i=1}^{nd} \lambda_i v_i^S (v_i^S)^T + \sum_{i=1}^{nd} \lambda_i v_i^S (v_i^S)^T + \sum_{i=1}^{nd} \lambda_i v_i^S (v_i^S)^T + \sum_{i=1}^{nd} \lambda_i v_i^S (v_i^S)^T. \]

On the other hand, we can decompose \( G + \Delta \) as

\[ G + \Delta = (G + P) + Q^1 + Q^2 + T, \]

(2.83)

where \( G + P \in S \otimes S, Q^1 \in S \otimes \bar{S}, Q^2 \in \bar{S} \otimes S \) and \( T \in S \otimes \bar{S} \). By comparing the right hand sides of (2.82) and (2.83), we conclude that

\[ G + P = \sum_{i=1}^{nd} \lambda_i v_i^S (v_i^S)^T, \quad T = \sum_{i=1}^{nd} \lambda_i v_i^S (v_i^S)^T, \]

\[ Q^1 = \sum_{i=1}^{nd} \lambda_i v_i^S (v_i^S)^T, \quad Q^2 = \sum_{i=1}^{nd} \lambda_i v_i^S (v_i^S)^T = (Q^1)^T. \]

Therefore, we have

\[
\|Q\|^2 = 2 \|Q^1\|^2 = 2 \left\| \sum_{i=1}^{nd} \lambda_i v_i^S (v_i^S)^T \right\|^2
\]

\[
= 2 \text{Tr} \left( \sum_{i=1}^{nd} \lambda_i v_i^S (v_i^S)^T \sum_{j=1}^{nd} \lambda_j v_j^S (v_j^S)^T \right)
\]

\[ = 2 \sum_{i,j=1}^{nd} \lambda_i \lambda_j (v_i^S)^T v_j^S (v_j^S)^T v_i^S \]

\[ \leq \sum_{i,j=1}^{nd} \lambda_i \lambda_j \left( (v_i^S)^T v_j^S (v_i^S)^T v_j^S + (v_j^S)^T v_i^S (v_j^S)^T v_i^S \right) \]

\[ = \text{Tr} \left( \sum_{i=1}^{nd} \lambda_i v_i^S (v_i^S)^T \sum_{j=1}^{nd} \lambda_j v_j^S (v_j^S)^T \right) + \text{Tr} \left( \sum_{i=1}^{nd} \lambda_i v_i^S (v_i^S)^T \sum_{j=1}^{nd} \lambda_j v_j^S (v_j^S)^T \right) \]

\[ = \|G + P\|^2 + \|T\|^2. \]

(2.84)
Using the fact that $G = I_{nd}$ and $nP_{11} = -\sum_i T_{ii}$, we obtain

$$
\|G + P\|^2 = \|G\|^2 + \|P\|^2 + 2\text{Tr}\langle G, P \rangle
= nd + \|P\|^2 - 2\text{Tr}\,(T). 
$$

(2.85)

Combining (2.79), (2.80), (2.85) and (2.84), we get (2.81).

Using Lemma 2.4.2 and Lemma 2.4.3, we reach the conclusion in the theorem.

**Remark** Using Lemma 2.4.3, $\|Q\|^2 \leq O(n^2)\epsilon^2$ holds when $\epsilon \gg 1/\sqrt{n}$, which leads to the weak stability result of LUD stated in Theorem 2.4.1 that requires $\epsilon$ to be fixed. In fact, a stronger stability result of LUD that allows $\epsilon \to 0$ as $n \to \infty$ can also be proven using ideas similar to the proof of Theorem 2.3.1.

### 2.4.2 Strong Stability

**Theorem 2.4.4** *(Strong stability)* Assume that all pairwise ratio measurements $R_{ij}$ are generated according to (2.71) such that the condition (2.72) holds for arbitrary small $\epsilon > 0$. Then there exists a critical probability $p_c^*(d)$ such that when $p > p_c^*(d)$, the solution $\hat{G}$ to the optimization problem (2.12) is close to the true Gram matrix $G$ in the sense that

$$
\|G - \hat{G}\|^2 \leq O(n^2)\epsilon^2
$$

w.h.p. (as $n \to \infty$). Moreover, an upper bound $p_c(d)$ for $p_c^*(d)$ is given by (2.14).

**Proof** To prove strong stability of LUD, we need a tighter lower bound for $f_2$. First we define the loss from the diagonal entries $f_2^d$ and that from the off-diagonal entries
We decompose $f_{2}^{\text{off}}$ as

\[
f_{2}^{\text{d}} = \sum_{(i,j) \in \mathcal{E}_c} \left( \left\| (I_d - R_{ij} + \Delta_{ij})^d \right\| - \left\| (I_d - R_{ij})^d \right\| \right) \]

\[
f_{2}^{\text{off}} = \sum_{(i,j) \in \mathcal{E}_c} \left( \left\| (I_d - R_{ij} + \Delta_{ij})^{\text{off}} \right\| - \left\| (I_d - R_{ij})^{\text{off}} \right\| \right).
\]

Then we have

\[
f_{2}^{\text{d}} \geq \sum_{(i,j) \in \mathcal{E}_c} \left( \left\| \Delta_{ij}^d \right\| - 2 \left\| (I_d - R_{ij})^d \right\| \right) \geq \left( \frac{pn}{\sqrt{d}} - O_{P} \left( \sqrt{n \log n} \right) \right) \text{Tr}(T) - 2pn^2\epsilon^2, \quad (2.86)
\]

since $\left\| (I_d - R_{ij})^d \right\| \approx \epsilon^2$. If $\text{Tr}(T) = O(n\epsilon^2)$, then using Lemma 2.4.3 we can show that $\|Q\|^2 = O(n^2\epsilon^2)$. And using (2.77) we are done. Thus we will continue with the case when $\text{Tr}(T) \gg n\epsilon^2$, where the term $2pn^2\epsilon^2$ in (2.86) is negligible. Now let us consider $f_{2}^{\text{off}}$. We further decompose $f_{2}^{\text{off}}$ to two parts as following

\[
f_{2}^{\text{off}} = \sum_{(i,j) \in \mathcal{E}_c} \left\langle \frac{(I_d - R_{ij})^{\text{off}}}{\left\| (I_d - R_{ij})^{\text{off}} \right\|}, \Delta_{ij}^{\text{off}} \right\rangle + \sum_{(i,j) \in \mathcal{E}_c} l(\Delta_{ij}), \quad (2.87)
\]

where

\[
l(\Delta_{ij}) := \left\| (I_d - R_{ij} + \Delta_{ij})^{\text{off}} \right\| - \left\| (I_d - R_{ij})^{\text{off}} \right\| - \left\langle \frac{(I_d - R_{ij})^{\text{off}}}{\left\| (I_d - R_{ij})^{\text{off}} \right\|}, \Delta_{ij}^{\text{off}} \right\rangle \geq 0.
\]
Apply the same analysis as that in section 2.3.2 to \[ \sum_{(i,j) \in E_c} \left\langle \frac{(I_d - R_{ij})^{\text{ff}}}{\| (I_d - R_{ij})^{\text{ff}} \|}, \Delta_{ij}^{\text{ff}} \right\rangle \] and notice that \[ \mathbb{E} \left( \frac{(I_d - R_{ij})^{\text{ff}}}{\| (I_d - R_{ij})^{\text{ff}} \|} \right) = 0, \] we obtain

\[
\sum_{(i,j) \in E_c} \left\langle \frac{(I_d - R_{ij})^{\text{ff}}}{\| (I_d - R_{ij})^{\text{ff}} \|}, \Delta_{ij}^{\text{ff}} \right\rangle \\
\geq - \left( \frac{1}{\sqrt{2}} \sqrt{1 - pm} + O_P \left( \sqrt{\log n} \right) \right) \text{Tr} (T) - O_P \left( \sqrt{n \log n} \right) \sum_i \| Q_i^{ss} \|. \tag{2.88}
\]

For the part \[ \sum_{(i,j) \in E_c} l (\Delta_{ij}) \], we have

\[
\sum_{(i,j) \in E_c} l (\Delta_{ij}) = \sum_{(i,j) \in E_c} l (P_{ij} + Q_{ij}^{ss} + Q_{ij}^{ss} + T_{ij}) \\
\geq \sum_{(i,j) \in E_c} l (Q_{ij}^{ss}) - 2 \sum_{(i,j) \in E_c} (\| P_{ij}^{\text{ff}} \| + \| T_{ij}^{\text{ff}} \| + \| Q_i^{ss} \|) \\
\geq \sum_{(i,j) \in E_c} l (Q_{ij}^{ss}) - c_8 n \text{Tr} (T). \tag{2.89}
\]

Now we consider

\[
\sum_{(i,j) \in E_c} l (Q_{ij}^{ss}) \\
= \sum_{(i,j) \in E_c} \left( \| (I_d - R_{ij} + Q_{ij}^{ss})^{\text{ff}} \| - \| (I_d - R_{ij})^{\text{ff}} \| - \left\langle \frac{(I_d - R_{ij})^{\text{ff}}}{\| (I_d - R_{ij})^{\text{ff}} \|}, Q_{ij}^{ss} \right\rangle \right),
\]

where \[ l (Q_{ij}^{ss}) \geq 0 \] and

\[
Q_{ij}^{ss} = Q_{ij}^{1,ss} + Q_{ij}^{2,ss} = Q_{ij}^{1,ss} - Q_{ij}^{1,ss}. \tag{2.90}
\]

Define two sets as

\[ S_1 = \left\{ i = 1, 2, \ldots, n \mid \| Q_i^{1,ss} \| \gg \epsilon \right\} \]
and

\[ S_2 = \{ i = 1, 2, \ldots, n | \| Q_i^{1,ss} \| = O(\epsilon) \}, \]

then from (2.90) we obtain

\[ Q_{ij}^{ss} \approx Q_{ij}^{1,ss}, \text{ if } i \in S_1 \text{ and } j \in S_2. \]

The set \( S_1 \) is assumed to be not empty, otherwise it is easy to see \( \| Q \|^2 = O_P(n^2\epsilon^2) \) and thus \( \| \Delta \|^2 = O_P(n^2\epsilon^2) \). In addition, \( \#S_1 \ll n \) since \( \sum_i \| Q_i^{1,ss} \| = O(n\epsilon) \).

For every \( i, \# \left\{ j | (i, j) \in E_c, \left( \frac{(I_d - R_{ij})^{off}}{\| (I_d - R_{ij})^{off} \|}, Q_i^{1,ss} \right) < 0 \right\} \geq c_9pn \), thus for every \( i \in S_1 \)

we have

\[ \# \left\{ j | (i, j) \in E_c, j \in S_2, \left( \frac{(I_d - R_{ij})^{off}}{\| (I_d - R_{ij})^{off} \|}, Q_i^{1,ss} \right) < 0 \right\} \geq c_9pn - \#S_1 \geq c_{10}pn. \]

Thus

\[
\sum_{(i,j) \in E_c} l(Q_{ij}^{ss}) \geq 2 \sum_{(i,j) \in E_c \cap (S_1 \times S_2)} l(Q_{ij}^{ss}) \\
\geq 2 \sum_{i \in S_1} \sum_{j \in S_2} \sum_{(i,j) \in E_c} \left( \frac{(I_d - R_{ij})^{off}}{\| (I_d - R_{ij})^{off} \|}, Q_i^{1,ss} \right) < 0 \right\} l(Q_{ij}^{ss}) \\
\geq 2 \sum_{i \in S_1} c_{11}pn \| Q_i^{1,ss} \| = c_{12}n \sum_{i \in S_1} \| Q_i^{1,ss} \|. \quad (2.91)
\]

Combine (2.87), (2.88), (2.89) and (2.91) together and we obtain

\[
f_2^{off} \geq c_{12}n \sum_{i \in S_1} \| Q_i^{1,ss} \| - O_P \left( \sqrt{n \log n} \right) \sum_i \| Q_i^1 \| - c_{13}n \text{Tr}(T). \quad (2.92)
\]
Apply Lemma 2.3.12 to (2.86) and (2.92), and set $\alpha = \sqrt{1 - \epsilon_0}$ and $\beta = \sqrt{\epsilon_0}$, where $\epsilon_0 \ll p - p_c$, then we obtain

$$
f_2 \geq \sqrt{1 - \epsilon_0} f_2^d + \sqrt{\epsilon_0} f_2^{off}
\geq \sqrt{1 - \epsilon_0} \left( \frac{pn}{\sqrt{d}} - \mathcal{O}_P \left( \sqrt{n \log n} \right) \right) \text{Tr} (T)
\quad + \sqrt{\epsilon_0} \left( c_{12} n \sum_{i \in S_1} \|Q_{i,ss}^1\| - \mathcal{O}_P \left( \sqrt{n \log n} \right) \sum_i \|Q_{ii}^1\| - c_{13} n \text{Tr} (T) \right).
$$

If $f_1 + f_2 \geq 0$, then $G + \Delta$ is not the minimizer. And $f_1 + f_2 \geq 0$ leads to the condition that

$$n \sum_{i \in S_1} \|Q_{i,ss}^1\| \geq \mathcal{O}_P \left( \sqrt{n} \right) \sum_i \|Q_{ii}^1\|.$$

Thus if $G + \Delta$ is the minimizer, then $\Delta$ must satisfy the condition that

$$n \sum_{i \in S_1} \|Q_{i,ss}^1\| \leq \mathcal{O}_P \left( \sqrt{n} \right) \sum_i \|Q_{ii}^1\| \leq \mathcal{O}_P \left( n \sqrt{n} \right) \epsilon,$$

that is,

$$\sum_{i \in S_1} \|Q_{i,ss}^1\| \leq \mathcal{O}_P \left( \sqrt{n} \right) \epsilon$$

therefore

$$\sum_i \|Q_{i,ss}^1\|^2 = \sum_{i \in S_1} \|Q_{i,ss}^1\|^2 + \sum_{i \in S_2} \|Q_{i,ss}^1\|^2
\quad \leq \left( \sum_{i \in S_1} \|Q_{i,ss}^1\| \right)^2 + \sum_{i \in S_2} \|Q_{i,ss}^1\|^2
\quad \leq \mathcal{O}_P \left( n \right) \epsilon^2 + n \mathcal{O}_P \left( \epsilon^2 \right) = \mathcal{O}_P \left( n \right) \epsilon^2$$

This, together with the decomposition [2.77] and arguments similar to [2.54] - [2.57], finishes the proof.
2.5 A Generalization of LUD to Random Incomplete Measurement Graphs

The analysis of exact and stable recovery of rotations from full measurements (section 2.3 and 2.4) can be straightforwardly generalized to the case of random incomplete measurement graphs. Here we assume that the edge set $\mathcal{E}$, which is the index set of measured rotation ratios $R_{ij}$, is a realization of a random graph drawn from the Erdős-Rényi model $\mathcal{G}(n, p_1), p_1 \geq 2 \log(n)/n$, and the rotation measurements $R_{ij}$ in the edge set $\mathcal{E}$ are generated according to (2.13) or (2.71). The reason why we have the restriction that $p_1 \geq 2 \log(n)/n$ is that as $n$ tends to infinity, the probability that a graph on $n$ vertices with edge probability $2 \log(n)/n$ is connected, tends to 1. The “good” edge set $\mathcal{E}_c$ and the “bad” edge set $\mathcal{E}\setminus\mathcal{E}_c$ can be seen as realizations of random graphs drawn from the Erdős-Rényi models $\mathcal{G}(n, p_1(1-p))$ and $\mathcal{G}(n, p_1p)$, respectively. As a consequence, we can apply the same arguments in section 2.3 and 2.4 and obtain the following theorems that are analogous to Theorem 2.3.1, 2.4.1 and 2.4.4. The associated numerical results are provided in section 2.7.2.

**Theorem 2.5.1** Assume that the index set of measured rotation ratios $\mathcal{E}$ is a realization of a random graph drawn from the Erdős-Rényi model $\mathcal{G}(n, p_1), p_1 \geq 2 \log(n)/n$, and the rotation ratio measurements $R_{ij}$ in $\mathcal{E}$ are generated according to (2.13). Then there exists a critical probability $p_0^*(d, p_1)$ such that when $p > p_0^*(d, p_1)$, the Gram matrix $G$ is exactly recovered by the solution to the optimization problem (2.12) w.h.p. (as $n \to \infty$). Moreover, an upper bound $p_c(d, p_1)$ for $p_0^*(d, p_1)$ is

$$p_c(d, p_1) = 1 - \left( -c_1(d) + \sqrt{c_1(d)^2 + 8p_1 \left( c(d) + 2/\sqrt{d} \right) / \sqrt{d}} \right)^2, \quad (2.93)$$
where \(c(d)\) and \(c_1(d)\) are constants defined in (2.15) and (2.16). In particular, when \(p_1 = 1\), \(p_c(d, 1) = p_c(d)\) in (2.14).

**Theorem 2.5.2** (Weak stability) Assume that the index set of measurements \(E\) is generalized as Theorem 2.5.1, and the rotation ratio measurements \(R_{ij}\) in \(E\) are generated according to (2.71) such that the condition (2.72) holds for a fixed small \(\epsilon > 0\). Then there exists a critical probability \(p^*_c(d, p_1)\) such that when \(p > p^*_c(d, p_1)\), the solution \(\hat{G}\) to the optimization problem (2.12) is close to the true Gram matrix \(G\) in the sense that

\[
\|G - \hat{G}\|^2 \leq O(n^2) \epsilon^2
\]

w.h.p. (as \(n \to \infty\)). Moreover, an upper bound \(p_c(d, p_1)\) for \(p^*_c(d, p_1)\) is given by (2.93).

**Theorem 2.5.3** (Strong stability) Assume that the index set of measurements \(E\) is generalized as Theorem 2.5.1, and the rotation ratio measurements \(R_{ij}\) in \(E\) are generated according to (2.71) such that the condition (2.72) holds for an arbitrary small \(\epsilon > 0\). Then there exists a critical probability \(p^*_c(d, p_1)\) such that when \(p > p^*_c(d, p_1)\), the solution \(\hat{G}\) to the optimization problem (2.12) is close to the true Gram matrix \(G\) in the sense that

\[
\|G - \hat{G}\|^2 \leq O(n^2) \epsilon^2
\]

w.h.p. (as \(n \to \infty\)). Moreover, an upper bound \(p_c(d, p_1)\) for \(p^*_c(d, p_1)\) is given by (2.93).

### 2.6 Alternating Direction Augmented Lagrangian Method of Multipliers (ADMM)

Here we briefly describe the ADMM [109] to solve the non-smooth minimization problem (2.12). ADMM is a multiple-splitting algorithm that minimizes the dual
augmented Lagrangian function sequentially regarding the Lagrange multipliers, then the dual slack variables, and finally the primal variables in each step. In addition, in the minimization over a certain variable, the other variables are kept fixed. The optimization problem (2.12) can be written as

\[
\min_{X_{ij}, G \succeq 0} \sum_{i<j} \|X_{ij}\|, \quad \text{s.t.} \quad \mathcal{A}(G) = b, \ X_{ij} = R_{ij} - G_{ij},
\]

where the operator \( \mathcal{A} : \mathbb{R}^{nd \times nd} \to \mathbb{R}^{nd^2} \) is defined as

\[
\mathcal{A}(G) = (G_{pq})_{i=1,...,n, p,q=1,...,d}
\]

and the row vector \( b \in \mathbb{R}^{nd^2} \) is

\[
b = 1_n \otimes \left( \mathcal{X}_{(p=q)}(p, q) \right)_{p,q=1,...,d}.
\]

Since

\[
\max_{\theta_{ij}, y, W \succeq 0} \min_{X_{ij}, G} \sum_{i<j} (\|X_{ij}\| - \langle \theta_{ij}, X_{ij} - R_{ij} + G_{ij} \rangle) - \langle y, \mathcal{A}(G) - b \rangle - \langle G, W \rangle
\]

\[
\Leftrightarrow \max_{\theta_{ij}, y, W \succeq 0} \min_{X_{ij}, G} - \langle Q(\theta) + W + \mathcal{A}^*(y), G \rangle + yb^T + \sum_{i<j} (\|X_{ij}\| - \langle \theta_{ij}, X_{ij} \rangle + \langle \theta_{ij}, R_{ij} \rangle)
\]

(2.95)

where

\[
Q(\theta) = \frac{1}{2} \begin{pmatrix}
0 & \theta_{12} & \cdots & \theta_{1m} \\
\theta_{T12} & 0 & \cdots & \theta_{T2m} \\
\vdots & \vdots & \ddots & \vdots \\
\theta_{T1m} & \theta_{T2m} & \cdots & 0
\end{pmatrix}
\]

We want to first minimize the function over \( X_{ij} \) and \( G \) in (2.95). The rearrangement of terms in (2.95) enable us to minimize \(- \langle Q(\theta) + W + \mathcal{A}^*(y), G \rangle \) over \( G \) and minimize
\[ \| X_{ij} \| - \langle \theta_{ij}, X_{ij} \rangle \] over \( X_{ij}, i < j \) separately. To minimize \(-\langle Q(\theta) + W + \mathcal{A}^*(y), G \rangle\) over \( G \), the optimum value will be \(-\infty\) if \( Q(\theta) + W + \mathcal{A}^*(y) \neq 0 \). Therefore due to the dual feasibility \( Q(\theta) + W + \mathcal{A}^*(y) = 0 \) and the optimum value is zero. Since

\[
\| X_{ij} \| - \langle \theta_{ij}, X_{ij} \rangle = \| X_{ij} \| - \| \theta_{ij} \| \| X_{ij} / \| \theta_{ij} \| \| X_{ij} / \| X_{ij} \| \|
\]

(2.96)

\[
\geq \| X_{ij} \| (1 - \| \theta_{ij} \|) \text{,} \quad (2.97)
\]

to minimize \( \| X_{ij} \| - \langle \theta_{ij}, X_{ij} \rangle \) over \( X_{ij} \), if \( \| \theta_{ij} \| > 1 \), then let \( X_{ij} = \alpha \theta_{ij}, \alpha > 0 \) and then

from (2.96) \( \| X_{ij} \| - \langle \theta_{ij}, X_{ij} \rangle = \alpha \| \theta_{ij} \| (1 - \| \theta_{ij} \|) \) goes to \(-\infty\) if \( \alpha \) goes to \(+\infty\). Hence \( \| \theta_{ij} \| \leq 1 \) and from (2.97) we get the optimum value is zero where \( X_{ij} = 0 \). Therefore the dual problem is

\[
\min_{\theta_{ij}, y, W \geq 0} -yb^T - \sum_{i<j} \langle \theta_{ij}, R_{ij} \rangle \text{ s.t. } \| \theta_{ij} \| \leq 1, Q(\theta) + W + \mathcal{A}^*(y) = 0. \quad (2.98)
\]

The augmented Lagrangian function of the dual problem (2.98) is

\[
\mathcal{L}(y, \theta, W, G) = -yb^T - \sum_{i<j} \langle \theta_{ij}, R_{ij} \rangle + \langle Q(\theta) + W + \mathcal{A}^*(y), G \rangle
\]

(2.99)

\[
+ \frac{\mu}{2} \| Q(\theta) + W + \mathcal{A}^*(y) \|^2_F, \text{ s.t. } \| \theta_{ij} \| \leq 1,
\]

where \( \mu > 0 \) is a penalty parameter. Then we can devise an alternating direction method (ADMM) that minimizes (2.99) with respect to \( y, \theta, W, \text{ and } G \) in an alternating fashion, that is, given some initial guess \( y^0, \theta^0, W^0, \text{ and } G^0 \), the simplest ADMM
method solves the following three subproblems sequentially in each iteration:

\begin{align}
\mathbf{y}^{k+1} &= \arg \min_{\mathbf{y}} \mathcal{L} \left( \mathbf{y}, \theta^{k}, W^{k}, G^{k} \right) \quad (2.100) \\
\theta_{ij}^{k+1} &= \arg \min_{\theta_{ij}} \mathcal{L} \left( \mathbf{y}^{k+1}, \theta, W^{k}, G^{k} \right), \text{ s.t. } \|\theta_{ij}\| \leq 1 \quad (2.101) \\
W^{k+1} &= \arg \min_{W \geq 0} \mathcal{L} \left( \mathbf{y}^{k+1}, \theta^{k+1}, W, G^{k} \right), \quad (2.102)
\end{align}

and updates the Lagrange multiplier \( G \) by

\begin{align}
G^{k+1} &= G^{k} + \gamma \mu \left( Q \left( \theta^{k+1} \right) + W^{k+1} + A^{*} \left( \mathbf{y}^{k+1} \right) \right), \quad (2.103)
\end{align}

where \( \gamma \in \left( 0, \frac{1+\sqrt{5}}{2} \right) \) is an appropriately chosen step length.

To solve (2.100), set \( \nabla_{\mathbf{y}} \mathcal{L} = 0 \) and using \( AA^{*} = I \), we obtain

\begin{align}
\mathbf{y}^{k+1} &= -A \left( Q \left( \theta^{k} \right) + W^{k} \right) - \frac{1}{\mu} \left( A \left( G^{k} \right) - \mathbf{b} \right).
\end{align}

By rearrangement of terms of \( \mathcal{L} \), it is easy to see problem (2.101) is equivalent to

\begin{align}
\min_{\theta_{ij}} - \langle \theta_{ij}, R_{ij} \rangle + \frac{\mu}{2} \|\theta_{ij} - \Phi_{ij}\|_{F}^{2}, \text{ s.t. } \|\theta_{ij}\| \leq 1
\end{align}

where \( \Phi = W^{k} + A^{*} \left( \mathbf{y}^{k+1} \right) + \frac{1}{\mu} G^{k} \). And it can be further simplified as

\begin{align}
\min_{\theta_{ij}} \langle \theta_{ij}, \mu \Phi_{ij} - R_{ij} \rangle + \frac{\mu}{2} \|\theta_{ij}\|^{2}, \text{ s.t. } \|\theta_{ij}\| \leq 1
\end{align}

whose solution is

\begin{align}
\theta_{ij} = \begin{cases} 
\frac{1}{\mu} R_{ij} - \Phi_{ij} & \text{if } \left\| \frac{1}{\mu} R_{ij} - \Phi_{ij} \right\| \leq 1, \\
\frac{R_{ij} - \mu \Phi_{ij}}{\left\| R_{ij} - \mu \Phi_{ij} \right\|} & \text{otherwise},
\end{cases}
\end{align}

71
Problem (2.102) is equivalent to

\[
\min \| W - H^k \|_F^2, \text{ s.t. } W \succeq 0,
\]

where \( H^k = -Q (\theta^{k+1}) - A^* (y^{k+1}) - \frac{1}{\mu} G^k \). Hence we obtain the solution \( W^{k+1} = V_+ \Sigma_+ V_+^T \), where

\[
V \Sigma V = (V_+ V_-) \begin{pmatrix} \Sigma_+ & 0 \\ 0 & \Sigma_- \end{pmatrix} \begin{pmatrix} V_+^T \\ V_-^T \end{pmatrix} \tag{2.104}
\]

is the spectral decomposition of the matrix \( H^k \), and \( \Sigma_+ \) and \( \Sigma_- \) are the positive and negative eigenvalues of \( H^k \).

Following (2.103), we have

\[
G^{k+1} = (1 - \gamma) G^k + \gamma \mu (W^{k+1} - H^k).
\]

The convergence analysis and the practical issues related to how to take advantage of low-rank assumption of \( G \) in the eigenvalue decomposition performed at each iteration, strategies for adjusting the penalty parameter \( \mu \), the use of a step size \( \gamma \) for updating the primal variable \( X \) and termination rules using the in-feasibility measures are discussed in details in [109]. According to the convergence rate analysis of ADMM in [39], we need \( O(1/\delta) \) iterations to reach a \( \delta \) accuracy. At each iteration, the most time-consuming step of ADMM is the computation of the eigenvalue decomposition in (2.104). Fortunately, for the synchronization problem, the primal solution \( G \) is a low rank matrix (i.e. \( \text{rank}(G) = d \)). Moreover, since the optimal solution pair \((y, \theta, W, G)\) satisfies the complementary condition \( W G = 0 \), the matrices \( W \) and \( G \) share the same set of eigenvectors and the positive eigenvalues of \( G \) corresponds to zero eigenvalues of \( W \). Therefore, at \( k \)th iteration we only need to compute \( V_- \), the part corresponding to the negative eigenvalues of \( W^k \). Thus to take advantage of the low rank structure of \( G \), we use the Arnoldi iterations [2] to compute first few negative
eigenvectors of $W^k$. However, for the noisy case, the optimal solution $G$ may have rank greater than $d$, and also during the iterations the rank of the solution $G^k$ may increase. Correspondingly, during the iterations $W^k$ may have more than $d$ negative eigenvalues. Therefore it is impossible to decide ahead of time how many negative eigenvectors of $W^k$ are required. A heuristic that could work well in practice is to compute only eigenvectors whose eigenvalues are smaller than some small negative threshold epsilon, with the hope that the number of such eigenvectors would be $o(n)$, yet not effecting the convergence of the algorithm. The Arnoldi iterations require $O(n^3)$ operations if $O(n)$ eigenvalues need to be computed. However, when first few negative eigenvalues of $W_k$ are required, the time cost by the Arnoldi iterations will be much reduced.

2.7 Numerical Experiments

All numerical experiments were performed on a machine with 2 Intel(R) Xeon(R) CPUs X5570, each with 4 cores, running at 2.93 GHz. We simulated 100, 500 and 1000 rotations in the groups $SO(2)$ and $SO(3)$ respectively. The noise is added to the rotation ratio measurements according to the ER random graph model $G(n, p)$ (2.13) and the model (2.71) with small perturbations on “good” edge set in subsection 2.7.1 and subsection 2.7.1, respectively, where $n$ is total number of rotations, and $p$ is the proportion of good rotation ratio measurements.

We define the relative error of the estimated Gram matrix $\hat{G}$ as

$$\text{RE} \left( \hat{G}, G \right) = \frac{\| \hat{G} - G \|}{\| G \|}, \quad (2.105)$$

and the mean squared error (MSE) of the estimated rotation matrices $\hat{R}_1, \ldots, \hat{R}_n$ as

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} \left\| R_i - \hat{O}\hat{R}_i \right\|^2, \quad (2.106)$$
where \( \hat{O} \) is the optimal solution to the registration problem between the two sets of rotations \( \{R_1, \ldots, R_n\} \) and \( \{\hat{R}_1, \ldots, \hat{R}_n\} \) in the sense of minimizing the MSE. As shown in [86], there is a simple procedure to obtain both \( \hat{O} \) and the MSE from the singular value decomposition of the matrix \( \frac{1}{n} \sum_{i=1}^{n} \hat{R}_i R_i^T \). For each experiment with fixed \( d, n, p \) and \( \kappa \), we run 10 trials and record the mean of REs and MSEs.

We compare LUD to EIG, SDP [84] (using the SDP solver SDPLR [11]). EIG and SDP are two algorithms to solve the least squares problem in (2.1) with equal weights using spectral relaxation and semidefinite relaxation, respectively. LUD does not have advantage in the running time. In our experiments, the running time of LUD using ADMM is about 10 to 20 times slower than that of SDP, and it is hundreds times slower than EIG. We will focus on the comparison of the accuracy of the rotation recovery using the three algorithms.

2.7.1 Experiments with Full Measurements

E1: Exact Recovery by LUD

In this experiment, we use LUD to recover rotations in \( SO(2) \) and \( SO(3) \) with different values of \( n \) and \( p \) in the noise model (2.13). Table 2.1 shows that when \( n \) is large enough, the critical probability where the Gram matrix \( G \) can be exactly recovered is very close to \( p_c(2) \approx 0.4570 \), \( p_c(3) \approx 0.4912 \).

The comparison of the accuracy of the estimated rotations by EIG, SDP and LUD is shown in Tables 2.2, 2.3 and Figure 2.2 that demonstrate LUD outperforms EIG and SDP in terms of accuracy.
Table 2.1: The Relative Error (2.105) of the Gram matrix $\hat{G}$ obtained by LUD in $\mathbf{E1}$. The critical probability where the Gram matrix $G$ can be exactly recovered is upper bounded by $p_c(2) \approx 0.4570$, $p_c(3) \approx 0.4912$ when $n$ is large enough.
<table>
<thead>
<tr>
<th></th>
<th>0.7</th>
<th>0.6</th>
<th>0.5</th>
<th>0.4</th>
<th>0.3</th>
<th>0.2</th>
</tr>
</thead>
<tbody>
<tr>
<td>EIG</td>
<td>0.0064</td>
<td>0.0115</td>
<td>0.0222</td>
<td>0.0419</td>
<td>0.0998</td>
<td>0.4285</td>
</tr>
<tr>
<td>SDP</td>
<td>0.0065</td>
<td>0.0116</td>
<td>0.0225</td>
<td>0.0427</td>
<td>0.1014</td>
<td>0.3971</td>
</tr>
<tr>
<td>LUD</td>
<td>1.7e-07</td>
<td>4.7e-08</td>
<td>8.4e-05</td>
<td>0.0043</td>
<td>0.0374</td>
<td>0.3296</td>
</tr>
</tbody>
</table>

(a) $n = 100$

<table>
<thead>
<tr>
<th></th>
<th>0.7</th>
<th>0.6</th>
<th>0.5</th>
<th>0.4</th>
<th>0.3</th>
<th>0.2</th>
</tr>
</thead>
<tbody>
<tr>
<td>EIG</td>
<td>0.0012</td>
<td>0.0023</td>
<td>0.0040</td>
<td>0.0077</td>
<td>0.0163</td>
<td>0.0445</td>
</tr>
<tr>
<td>SDP</td>
<td>0.0012</td>
<td>0.0023</td>
<td>0.0041</td>
<td>0.0078</td>
<td>0.0164</td>
<td>0.0440</td>
</tr>
<tr>
<td>LUD</td>
<td>6.4e-10</td>
<td>5.5e-09</td>
<td>9.6e-09</td>
<td>6.3e-05</td>
<td>0.0025</td>
<td>0.0211</td>
</tr>
</tbody>
</table>

(b) $n = 500$

<table>
<thead>
<tr>
<th></th>
<th>0.7</th>
<th>0.6</th>
<th>0.5</th>
<th>0.4</th>
<th>0.3</th>
<th>0.2</th>
</tr>
</thead>
<tbody>
<tr>
<td>EIG</td>
<td>0.0006</td>
<td>0.0011</td>
<td>0.0020</td>
<td>0.0037</td>
<td>0.0080</td>
<td>0.0207</td>
</tr>
<tr>
<td>SDP</td>
<td>0.0006</td>
<td>0.0011</td>
<td>0.0020</td>
<td>0.0037</td>
<td>0.0081</td>
<td>0.0207</td>
</tr>
<tr>
<td>LUD</td>
<td>3.0e-10</td>
<td>1.5e-09</td>
<td>7.3e-09</td>
<td>7.5e-06</td>
<td>0.0010</td>
<td>0.0084</td>
</tr>
</tbody>
</table>

(c) $n = 1000$

Table 2.2: MSE (3.73) of the estimated rotations in $SO(2)$ using EIG, SDP and LUD in $E_1$. The critical probability where the rotations can be exactly recovered is upper bounded by $p_c(2) \approx 0.4570$ when $n$ is large enough.
Table 2.3: MSE (3.73) of the estimated rotations in $SO(3)$ using EIG, SDP and LUD in $E_1$. The critical probability where the rotations can be exactly recovered is upper bounded by $p_c(3) \approx 0.4912$ when $n$ is large enough.
Figure 2.2: MSE [3.73] of the estimated rotations in \( SO(3) \) and \( SO(2) \) using EIG, SDP and LUD for different values of \( n \) and \( p \) in E1. Exact Recovery (i.e. \( \log_{10}(MSE) < -7 \)) is achieved by LUD.

**E2: Stability of LUD**

In this experiment, the three algorithms are used to recover rotations in \( SO(2) \) with different values of \( n, p \) in the noise model [2.71]. In [2.71], the perturbed rotations \( \bar{R}_{ij} \) for the “good” edge set are sampled from a von Mises-Fisher distribution [13] with mean rotation \( R_i^T R_j \) and a concentration parameter \( \kappa > 0 \). The probability density function of the von Mises-Fisher distribution for \( \bar{R}_{ij} \) is given by:

\[
   f \left( \bar{R}_{ij}; R_i^T R_j, \kappa \right) = c(\kappa) \exp \left( \kappa \text{Tr} \left( R_j^T R_i \bar{R}_{ij} \right) \right),
\]

where \( c(\kappa) \) is a normalization constant. The parameters \( R_i^T R_j \) and \( 1/\kappa \) are analogous to \( \mu \) (mean) and \( \sigma^2 \) (variance) of the normal distribution:
1. $R_i^T R_j$ is a measure of location (the distribution is clustered around $R_i^T R_j$).

2. $\kappa$ is a measure of concentration (a reciprocal measure of dispersion, so $1/\kappa$ is analogous to $\sigma^2$). If $\kappa$ is zero, the distribution is uniform, and for small $\kappa$, it is close to uniform. If $\kappa$ is large, the distribution becomes very concentrated about the rotation $R_i^T R_j$. In fact, as $\kappa$ increases, the distribution approaches a normal distribution in $R_{ij}$ with mean $R_i^T R_j$ and variance $1/\kappa$.

For arbitrary fixed small $\epsilon > 0$, we can choose the concentration parameter $\kappa$ large enough so that the condition (2.72) is satisfied. In fact, using Weyl integration formula (2.45), it can be shown that when $\kappa \to \infty$,

$$
\epsilon = \mathbb{E}(\| \bar{R}_{ij} - R_i^T R_j \|) = \sqrt{\frac{2}{\pi \kappa}} + O(\kappa^{-3/2}),
$$

$$
\text{Var}(\| \bar{R}_{ij} - R_i^T R_j \|) = \left(1 - \frac{2}{\pi}\right) \frac{1}{\kappa} + O(\kappa^{-2}).
$$

Figures 2.3a, 2.3b and 2.4 show that for different $n$, $p$ and concentration parameter $\kappa$, LUD is more accurate than EIG and SDP. In Figure 2.4, The MSEs of LUD, EIG and SDP are compared with the Cramér-Rao bound (CRB) for synchronization. In [7], Boumal et. al. established formulas for the Fisher information matrix and associated CRBs that are structured by the pseudoniverse of the Laplacian of the measurement graph.

In addition we observe that when the concentration parameter $\kappa$ is as large as 100, which means the perturbations on the “good” edges are small, the critical probability $p_c$ for phase transition can be clearly identified around 0.5. As $\kappa$ decreases, the phase transition becomes less obvious.
Figure 2.3: MSE (3.73) of the estimated rotations in \( SO(2) \) using EIG, SDP and LUD for 100 and 500 rotations and different values of \( p \) and concentration parameter \( \kappa \) in \( E_2 \). LUD is stable when there are small perturbations on the “good” edge set.

Figure 2.4: MSE of \( n = 100 \) estimated rotations in \( SO(2) \) as a function of \( \kappa \) when \( p = 0.7 \) using LUD, EIG and SDP. The recovery by LUD is stable to small perturbations on the “good” edges as indicated by the linear relationship of \( \log(\text{MSE}) \) and \( \log(\kappa) \). The green dotted line represents the Cramér-Rao bound for synchronization [7].
2.7.2 Experiments with Incomplete Measurements

In the experiments E3 and E4 shown in Figure 2.5 and 2.6, the measurements $R_{ij}$ are generated as in experiments E1 and E2, respectively, with the exception that instead of using the complete graph of measurements as in E1 and E2, the index set of measurements $\mathcal{E}$ is a realization of a random graph drawn from the Erdős-Rényi model $\mathcal{G}(n, p_1)$, where $p_1$ is the proportion of measured rotation ratios. The results demonstrate the exact recovery and stability of LUD with incomplete measurements that are described in Section 2.5.

![Figure 2.5: Phase transitions of LUD on incomplete graphs (E3). The color intensity of each pixel represents log(MSE), depending on the edge probability $p_1$ (x-axis), and the “good” edge probability $p$ (y-axis). The blue curves are the upper bounds of the critical probability $p_c(d, p_1)$ in Theorem 2.5.1. Both experiments used $n = 500$ rotations.](image-url)
Figure 2.6: MSE of $n = 500$ estimated rotations in $SO(2)$ as a function of $\kappa$ with different values of the edge probability $p_1$ and the “good” edge probability $p$ using LUD, EIG and SDP (\textbf{E4}). The recovery by LUD is stable to small perturbations on the “good” edges as indicated by the linear relationship of $\log(MSE)$ and $\log(\kappa)$. The green dotted line represents the Cramér-Rao bound for synchronization [7].

2.7.3 Real Data Experiments

We tried LUD for solving the global alignment problem of 3D scans from the Lucy dataset\footnote{Available from The Stanford 3-D Scanning Repository at http://www-graphics.stanford.edu/data/3Dscanrep/} (see Figure 2.7). We are using a down-sampled version of the dataset containing 368 scans with a total number of 3.5 million triangles. Running the automatic Iterative Closest Point (ICP) algorithm [77] starting from initial estimates returned 2006 pairwise transformations. For this model, we only have the best reconstruction found so far at our disposal but no actual ground truth. Nevertheless, we use this reconstruction to evaluate the error of the estimated rotations.

We apply the two algorithms LUD, EIG on the Lucy dataset since we observed SDP did not perform so well on this dataset. Although the MSEs are quite similar (0.4044 for LUD and 0.3938 for EIG), we observe that the unsquared residuals $\|\hat{R}_i - R_i\|$ ($i = 1, 2, \ldots, 368$), where $\hat{R}_i$ is the estimated rotation, are more concentrated around zero for LUD (Figure 2.8). Figure 2.9 suggests that the “bad” edges $(i, j)$ (edges with truly large measurement errors in the left subfigure of Figure 2.9) can be eliminated using the results of LUD more robustly, compared to that of EIG. We set the cutoff value to be 0.1 in Figure 2.9 for the estimated measurement errors.
obtained by LUD and EIG. Then 1527 and 1040 edges are retained from 2006 edges by LUD and EIG respectively, and the largest connected component of the sparsified graph (after eliminating the seemingly “bad” edges) has size 312 and 299 respectively. The 3D scans with the estimated rotations in the largest component are used in the reconstruction. The reconstruction obtained by LUD is better than that by EIG (Figure 2.10).

Figure 2.7: The Lucy statue.

Figure 2.8: Histogram of the unsquared residuals of EIG, SDP and LUD for the Lucy dataset. The errors by LUD are more concentrated near zero.
Figure 2.9: True and estimated rotation ratio measurement errors for the Lucy dataset. The measurements are sparse since only 2006 rotation ratios were measured for 368 3-D scans. Left: the value at the \((i, j)\)th element is \(\|R_{ij} - R_i^T R_j\|\) if it is non-empty, where \(i, j = 1, 2, \ldots, 368\). Middle and Right: similar to Left except that the value at the \((i, j)\)th element is \(\|\hat{R}_{ij} - \hat{R}_i^T \hat{R}_j\|\) if it is non-empty.
Figure 2.10: Front and back views of the reconstructions from the Lucy dataset. The left one is the best found reconstruction. The middle and right ones are obtained by LUD and EIG.

2.8 Summary

In this chapter we proposed to estimate the rotations using LUD. LUD minimizes a robust self consistency error, which is the sum of unsquared residuals instead of the sum of squared residuals. LUD is then semidefinite relaxed and solved by ADMM. We compare LUD method to EIG and SDP methods, both of which are based on least squares approach, and demonstrate that the results obtained by LUD are the
most accurate. When the noise in the rotation ratio measurements comes from ER random graph model $G(n,p)$, we compute an upper bound $p_c$ of the phase transition point such that the rotations can be exactly recovered when $p > p_c$. Moreover, the solution of LUD is stable when small perturbations are added to "good" rotation ratio measurements. We also showed exact recovery and stability for LUD when the measurements of the rotation ratios are incomplete and the measured rotation ratios come from ER random graph model $G(n,p_1)$.

The exact recovery result for the noise model (2.13) is actually not that surprising. In order to determine if the rotation measurement for a given edge $(i,j)$ is correct or noisy, we can consider all cycles of length three (triangles) that include that edge. There are $n - 2$ such triangles, and we can search for a consistent triangle by multiplying the three rotation ratios and checking if the product is the identity rotation. If there is such a consistent triangle, then the measurement for the edge $(i,j)$ is almost surely correct. The expected number of such consistent triangles is $(n - 2)p^2$, so the critical probability for such an algorithm is $p_c = O(1/\sqrt{n})$, which is already significantly better than our exact recovery condition for LUD for which the critical probability does not depend on $n$. In fact, by considering all cycles of length $k \geq 3$ consisting of a given fixed edge (there are $O(n^{k-2})$ such cycles), the expected number of consistent cycles is $O(n^{k-2}p^{k-1})$, therefore the critical probability is $O(1/n^{1-\varepsilon})$ for any $\varepsilon > 0$. Since exact recovery requires the graph of good edges is connected, and the ER graph is connected almost surely when \( p = \frac{2 \log n}{n} \), the critical probability cannot be smaller than \( p = \frac{2 \log n}{n} \). The computational complexity of cycle-based algorithms increases exponentially with the cycle length, but already for short cycles (e.g., $k = 3$) they are preferable to LUD in terms of the critical probability. So what is the advantage of LUD compared to cycle-based algorithms? The answer to this question lies in our stability result. While LUD is stable to small perturbations on
the good edges, cycle-based algorithms are unstable, and are therefore not as useful in applications where small measurement noise is present.

An iterative algorithm for robust estimation of rotations has been recently proposed in [38] for applications in computer vision. That algorithm aims to minimize the sum of geodesic distances between the measured rotation ratios and those derived from the estimated rotations. In each iteration, the algorithm sequentially updates the rotations by the median of their neighboring rotations using the Weiszfeld algorithm. We tested this algorithm numerically and find it to perform well, in the sense that its critical probability for exact recovery for the noise model (2.13) is typically smaller than that of LUD. However, the objective function that algorithm aims to minimize is not even locally convex. It therefore requires a good initial guess which can be provided by either EIG, SDP or LUD. In a sense, that algorithm may be regarded as a non-convex refinement algorithm for the estimated rotations. There is currently no theoretical analysis of that algorithm due to the non-convexity and the non-smoothness of its underlying objective function.

In the future, we plan to extend the LUD framework in at least two ways. First, we plan to investigate exact and stable recovery conditions for more general (and possibly weighted) measurement graphs other than the complete and random ER graphs considered here. We speculate that the second eigenvalue of the graph Laplacian would play a role in conditions for exact and stable recovery, similar to the role it plays in the bounds for synchronization obtained in [6]. Second, the problem of synchronization of rotations is closely related to the orthogonal Procrustes problem of rotating $n$ matrices toward a best least-squares fit [95]. Closed form solution is available only for $n = 2$, and a certain SDP relaxation for $n \geq 3$ was analyzed in [60]. The LUD framework presented here can be extended in a straightforward manner for the orthogonal Procrustes problem of rotating $n$ matrices toward a best least-unsquared deviations fit, which could be referred to as the robust orthogonal Procrustes prob-
lem. Similar to the measurement graph assumed in our theoretical analysis of LUD for robust synchronization, also in the Procrustes problem the measurement graph is the complete graph (all pairs of matrices are compared).
Chapter 3

Orientation Determination of Cryo-EM images Using Least Unsquared Deviations

A major challenge in single particle reconstruction from cryo-electron microscopy is to establish a reliable ab-initio three-dimensional model using two-dimensional projection images with unknown orientations. Common-lines based methods estimate the orientations without additional geometric information. However, such methods fail when the detection rate of common-lines is too low due to the high level of noise in the images. In this chapter, we apply LUD method to improve the accuracy of orientation determination. The LUD problem is then semidefinite relaxed and solved via ADMM or IRLS procedure. In section 3.1 we review the detection procedure of common lines between images. Section 3.2 presents the LS and LUD global self-consistency cost functions. Section 3.3 introduces the semidefinite relaxation and rounding procedure for the LUD formulation. The additional spectral norm constraint to tighten the relaxation is considered in section 3.4. The ADMM method for obtaining the global minimizer is detailed in Section 3.5 and the IRLS procedure is described in section
Numerical results for both simulated and real data are provided in section 3.7. Finally, section 3.8 is a summary.

### 3.1 Detection of Common Lines Between Images

Recall that according to the Fourier projection-slice theorem (Figure 1.1), we can relate different projections of the same object to one another. In Fourier space, 2D projections are represented by central planes, which intersect each other along lines that go through the origin: the common lines. The common lines between any three projection images with linearly independent projection directions provide some non-trivial information about their relative orientations, i.e., they uniquely determine their relative orientations up to handedness (chirality). Conversely, the common lines can be used to determine the orientations of projections.

Typically, the first step for detecting common lines is to compute the 2D Fourier transform of each image on a polar grid using, e.g., the non-uniform fast Fourier transform (NUFFT) \cite{21,24,32}. The transformed images have resolution \( n_r \) in the radial direction and resolution \( n_\theta \) in the angular direction, that is, the radial resolution \( n_r \) is the number of equi-spaced samples along each ray in the radial direction, and the angular resolution \( n_\theta \) is the number of angularly equally-spaced Fourier rays computed for each image (Figure 1.1). For simplicity, we let \( n_\theta \) be an even number. The transformed images are denoted as \( \{\vec{l}_k^0, \vec{l}_k^1, \ldots, \vec{l}_k^{n_\theta-1}\} \), where \( \vec{l}_m^k = (l_{m,1}^k, l_{m,2}^k, \ldots, l_{m,n_r}^k) \) is an \( n_r \) dimensional vector, \( m \in \{0, 1, \ldots, n_\theta - 1\} \) is the index of a ray, \( k \in \{1, 2, \ldots, K\} \) is the index of an image and \( K \) is the number of images. The DC term is shared by all lines independently of the image, and is therefore excluded for comparison. To determine the common line between two images \( P_i \) and \( P_j \), the similarity between all \( n_\theta \) radial lines \( \vec{l}_0^i, \vec{l}_1^i, \ldots, \vec{l}_{n_\theta-1}^i \) from the first image with all \( n_\theta \) radial lines \( \vec{l}_0^j, \vec{l}_1^j, \ldots, \vec{l}_{n_\theta-1}^j \) from the second image are measured (overall \( n_\theta^2 \) comparisons), and the pair of radial
lines $\vec{l}_{m_{i,j}}$ and $\vec{l}_{m_{j,i}}$ with the highest similarity is declared as the common-line pair between the two images. However, as a radial line is the complex conjugate of its antipodal line, the similarity measure between $\vec{l}_{m_1}$ and $\vec{l}_{m_2}$ has the same value as that between their antipodal lines $\vec{l}_{m_1+n_\theta/2}$ and $\vec{l}_{m_2+n_\theta/2}$ (where addition of indices is taken modulo $n_\theta$). Thus the number of distinct similarity measures that need to be computed is $n_\theta^2/2$ obtained by restricting the index $m_1$ to take values between 0 and $n_\theta/2$ and letting $m_2$ take any of the $n_\theta$ possibilities (see also [101] and [65], p. 255). Equivalently, it is possible to compare real valued 1D line projections of the 2D projection images, instead of comparing radial Fourier lines that are complex valued. According to the Fourier projection-slice theorem, each 1D projection is obtained by the inverse Fourier transform of the corresponding Fourier radial line $\vec{l}_m^k$ and its antipodal line $\vec{l}_{m+n_\theta/2}^k$, and is denoted as $\vec{s}_m^k$. The 1D projection lines of a cryo-EM image can be displayed as a 2D image known as a “sinogram” (see [101, 82]).

Traditionally, the pair of radial lines (or sinogram lines) that has the maximum normalized cross correlation is declared as the common line, that is,

$$\left( m_{i,j}^*, m_{j,i}^* \right) = \arg \max_{0 \leq m_1 < n_\theta/2, 0 \leq m_2 < n_\theta} \left\{ \frac{\vec{l}_{m_1} \cdot \vec{l}_{m_2}}{\| \vec{l}_{m_1} \| \cdot \| \vec{l}_{m_2} \|} \right\}, \text{ for all } i \neq j, \quad (3.1)$$

where $m_{i,j}$ is a discrete estimate for where the $j$th image intersects with the $i$th image. In practice, a weighted correlation, which is equivalent to applying a combination of high-pass and low-pass filters is used to determine proximity. As noted in [101], the normalization is performed so that the correlation coefficient becomes a more reliable measure of similarity between radial lines. Note that even with clean images, this estimate will have a small deviation from its ground truth (unknown) value due to discretization errors. With noisy images, large deviations of the estimates from their true values (say, errors of more than $10^\circ$) are frequent, and their frequency increases with the level of noise. We refer to common lines whose $m_{i,j}$ and
\( m_{j,i} \) values were estimated accurately (up to a given discretization error tolerance) as “correctly detected” common lines, or “inliers” and to the remaining common lines as “falsely detected”, or “outliers”.

### 3.2 Weighted LS and LUD

We define the directions of detected common-lines between the transformed image \( i \) and transformed image \( j \) as unit vectors (Figure 1.1)

\[
\vec{c}_{ij} = (c^1_{ij}, c^2_{ij}) = (\cos(2\pi m_{ij}/n\theta), \sin(2\pi m_{ij}/n\theta)), \quad (3.2)
\]

\[
\vec{c}_{ji} = (c^1_{ji}, c^2_{ji}) = (\cos(2\pi m_{ji}/n\theta), \sin(2\pi m_{ji}/n\theta)), \quad (3.3)
\]

where \( \vec{c}_{ij} \) and \( \vec{c}_{ji} \) are on the transformed images \( i \) and \( j \) respectively, and \( m_{ij} \) and \( m_{ji} \) are discrete estimate for the common lines’ positions using (3.1). Let the rotation matrices \( R_i \in SO(3), i = 1, \cdots, K \) represent the orientations of the \( K \) images. According to the Fourier projection-slice theorem (Figure 1.1), the common lines on every two images should be the same after the 2D transformed images are inserted in the 3D Fourier space using the corresponding rotation matrices, that is,

\[
R_i \begin{pmatrix} \vec{c}_{ij}^T \\ 0 \end{pmatrix} = R_j \begin{pmatrix} \vec{c}_{ji}^T \\ 0 \end{pmatrix} \quad \text{for} \quad 1 \leq i < j \leq K. \quad (3.4)
\]

These can be viewed as \( \begin{pmatrix} K \\ 2 \end{pmatrix} \) linear equations for the \( 6K \) variables corresponding to the first two columns of the rotation matrices (the third column of each rotation matrix does not contribute in (3.4) due to the zero third entries in the common-line vectors in \( \mathbb{R}^3 \)). The weighted LS approach for solving this system can be formulated.
Figure 3.1: Left and Middle: The histogram plots of errors in the detected common-
lines $\vec{c}_{ij}$ for all $i$ and $j$, i.e., $\| R_i (\vec{c}_{ij}, 0)^T - R_j (\vec{c}_{ji}, 0)^T \|$ where $R_i$ is a true rotation matrix for all $i$. The fat tail in (b) indicates the detected common-lines contain a large amount of outliers. Right: elucidating the difference between the squared distance and the absolute deviation.

as the minimization problem

$$\min_{R_1, \ldots, R_K \in SO(3)} \sum_{i \neq j} w_{ij} \| R_i (\vec{c}_{ij}, 0)^T - R_j (\vec{c}_{ji}, 0)^T \|^2,$$

where the weights $w_{ij}$ indicate the confidence in the detections of common-lines between pairs of images. Since $(\vec{c}_{ij}, 0)^T$ and $(\vec{c}_{ji}, 0)^T$ are 3D unit vectors, their rotations are also unit vectors; that is, $\| R_i (\vec{c}_{ij}, 0)^T \| = \| R_j (\vec{c}_{ji}, 0)^T \| = 1$. It follows that the minimization problem (3.5) is equivalent to the maximization problem of the sum of dot products

$$\max_{R_1, \ldots, R_K \in SO(3)} \sum_{i \neq j} w_{ij} \langle R_i (\vec{c}_{ij}, 0)^T, R_j (\vec{c}_{ji}, 0)^T \rangle.$$

When the weight $w_{ij} = 1$ for each pair $i \neq j$, (3.6) is equivalent to the LS problem that was considered in [71], and more recently in [86] using convex relaxation of the non-convex constraint set. The solution to the LS problem may not be optimal due to the typically large proportion of outliers (Figure 3.1).

To guard the estimation of the orientations from outliers, we replace the sum of weighted squared residuals in (3.5) with the more robust sum of unsquared residuals
and obtain
\[
\min_{R_1, \ldots, R_K \in \text{SO}(3)} \sum_{i \neq j} \left\| R_i (\vec{c}_{ij}, 0)^T - R_j (\vec{c}_{ji}, 0)^T \right\|,
\tag{3.7}
\]
or equivalently,
\[
\min_{R_1, \ldots, R_K \in \text{SO}(3)} \sum_{i \neq j} \left\| (\vec{c}_{ij}, 0)^T - R_i^T R_j (\vec{c}_{ji}, 0)^T \right\|.
\tag{3.8}
\]
We refer to the minimization problem \ref{eq:least_unsquared_deviations} as the least unsquared deviations (LUD) problem in this chapter. The self consistency error given in \ref{eq:least_unsquared_deviations} reduces the contribution from large residuals that may result from outliers (Figure 3.1c). We remark that it is also possible to consider the weighted version of \ref{eq:least_unsquared_deviations}, namely
\[
\min_{R_1, \ldots, R_K \in \text{SO}(3)} \sum_{i \neq j} w_{ij} \left\| R_i (\vec{c}_{ij}, 0)^T - R_j (\vec{c}_{ji}, 0)^T \right\|.
\]
For simplicity, we focus here on the unweighted version.

### 3.3 Semidefinite Programming Relaxation (SDR) and the Rounding Procedure

Both the weighted LS problem \ref{eq:weighted_least_squares} and the LUD problem \ref{eq:least_unsquared_deviations} are non-convex and therefore extremely difficult to solve if one requires the matrices $R_i$ to be rotations, that is, when adding the constraints
\[
R_i R_i^T = I_3, \quad \det (R_i) = 1, \quad \text{for } i = 1, \ldots, K,
\tag{3.9}
\]
where $I_3$ is the $3 \times 3$ identity matrix. A relaxation method that neglects the constraints \ref{eq:rotation_constraints} will simply collapse to the trivial solution $R_1 = \ldots = R_K = 0$ which obviously does not satisfy the constraint \ref{eq:rotation_constraints}. 

94
The relaxation in [86] that uses semidefinite programming (SDP) can be modified in a straightforward manner in order to deal with non-unity weights $w_{ij}$ in (3.6). We present this modification here for three reasons. First, the weighted version is required by the IRLS procedure (see Section 3.6). Second, the rounding procedure after SDP employed here is slightly different than the one presented in [86] and is closer in spirit to the rounding procedure of Goemans and Williamson for the MAX-CUT problem [30]. Finally, in the section 3.3.5 we prove exact recovery of the rotations by the semidefinite relaxation procedure when the detected common-lines are all correct.

3.3.1 Constructing the Gram matrix $\tilde{G}$ from the Rotations $R_i$

We denote the columns of the rotation matrix $R_i$ by $R^1_i$, $R^2_i$, and $R^3_i$, and write the rotation matrices as

$$R_i = \begin{pmatrix} \vert & \vert & \vert \\ R^1_i & R^2_i & R^3_i \\ \vert & \vert & \vert \end{pmatrix}, \quad i = 1, \ldots, K.$$ 

We define a $3 \times 2K$ matrix $\tilde{R}$ by concatenating the first two columns of all rotation matrices:

$$\tilde{R} = \begin{pmatrix} \vert & \vert & \vert & \vert & \vert & \vert & \vert & \vert & \vert & \vert & \vert & \vert \\ R^1_1 & R^2_1 & \cdots & R^1_k & R^2_k & \cdots & R^1_K & R^2_K \\ \vert & \vert & \vert & \vert & \vert & \vert & \vert & \vert & \vert & \vert & \vert & \vert \end{pmatrix}.$$ 

(3.10)

The Gram matrix $\tilde{G}$ for the matrix $\tilde{R}$ is a $2K \times 2K$ matrix of inner products between the 3D column vectors of $\tilde{R}$, that is,

$$\tilde{G} = \tilde{R}^T \tilde{R}.$$ 

(3.11)
Clearly, $\tilde{G}$ is a rank-3 semidefinite positive matrix ($\tilde{G} \succeq 0$), which can be conveniently written as a block matrix

$$
\tilde{G} = \left( \tilde{G}_{ij} \right)_{i,j=1,\ldots,K},
$$

where $\tilde{G}_{ij}$ is the $2 \times 2$ upper left block of the rotation matrix $R_i^T R_j$, that is,

$$
\tilde{G}_{ij} = \begin{pmatrix}
(R_i^1)^T \\
(R_i^2)^T
\end{pmatrix}
\begin{pmatrix}
R_i^1 & R_i^2
\end{pmatrix}.
$$

In addition, the orthogonality of the rotation matrices ($R_i^T R_i = I$) implies that

$$
\tilde{G}_{ii} = I_2, \quad i = 1, 2, \ldots, K,
$$

where $I_2$ is the $2 \times 2$ identity matrix.

### 3.3.2 SDR for Weighted LS

We first define two $2K \times 2K$ matrices $S = (S_{ij})_{i,j=1,\ldots,K}$ and $W = (W_{ij})_{i,j=1,\ldots,K}$, where the $2 \times 2$ sub-blocks $S_{ij}$ and $W_{ij}$ are given by

$$
S_{ij} = \tilde{c}_j^T \tilde{c}_{ij},
$$

and

$$
W_{ij} = w_{ij} \begin{pmatrix}
1 & 1 \\
1 & 1
\end{pmatrix}.
$$

Both matrices $S$ and $W$ are symmetric and they store all available common-line information and weight information, respectively. It follows that the objective function
is the trace of the matrix $(W \circ S) \tilde{G}:
\sum_{i \neq j} w_{ij} \langle R_i (\vec{c}_{ij}, 0)^T, R_j (\vec{c}_{ji}, 0)^T \rangle = \text{trace} \left( (W \circ S) \tilde{G} \right),
(3.13)

where the symbol $\circ$ denotes the Hadamard product between two matrices. A natural relaxation of the optimization problem (3.6) is thus given by the SDP problem
\[
\max_{\tilde{G} \in \mathbb{R}^{2K \times 2K}} \text{trace} \left( (W \circ S) \tilde{G} \right)
\text{ s.t. } \tilde{G}_{ii} = I_2, \ i = 1, 2, \ldots, K,
\tilde{G} \succeq 0
(3.14)-(3.16)
\]

The non-convex rank-3 constraint on the Gram matrix $\tilde{G}$ is missing from this semidefinite relaxation (SDR) [50]. The problem (3.14)-(3.16) is an SDP that can be solved by standard SDP solvers. In particular, it can be well solved by the solver SDPLR [11] which takes advantage of the low-rank property of $\tilde{G}$. SDPLR is a first-order algorithm via low-rank factorization and hence can provide approximate solutions for large scale problems. Moreover, the iterations of SDPLR are extremely fast.

3.3.3 SDR for LUD

Similar to defining the Gram matrix $\tilde{G}$ in (3.11), we define a $3K \times 3K$ matrix $G$ as $G = (G_{ij})_{i,j=1,\ldots,K}$, where each $G_{ij}$ is a $3 \times 3$ block defined as $G_{ij} = R_i^T R_j$. Then, a natural SDR for (3.8) is given by
\[
\min_{G \succeq 0} \sum_{i \neq j} \left\| (\vec{c}_{ij}, 0)^T - G_{ij} (\vec{c}_{ji}, 0)^T \right\|, \text{ s.t. } G_{ii} = I_3.
(3.17)
\]

The constraints missing in this SDP formulation are the non-convex rank-3 constraint and the determinant constraints $\det(G_{ij}) = 1$ on the Gram matrix $G$. However, the
solution $G$ to (3.17) is not unique. Note that if a set of rotation matrices $\{R_i\}$ is the solution to (3.8), then the set of conjugated rotation matrices $\{JR_iJ\}$ is also the solution to (3.8), where the matrix $J$ is defined as

$$J = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix}.$$ 

Thus, another solution to (3.17) is the Gram matrix $G^J = (G^J_{ij})_{i,j=1,\ldots,K}$ with the $3 \times 3$ sub-blocks given by $G^J_{ij} = JR_i^T J R_j J = JR_i^T R_j J$. It can be verified that $\frac{1}{2}(G + G^J)$ is also a solution to (3.17). Using the fact that

$$\frac{1}{2}(G_{ij} + G^J_{ij}) = \begin{pmatrix} \tilde{G}_{ij} & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix},$$

the problem (3.17) is reduced to

$$\min_{\tilde{G} \succeq 0} \sum_{i \neq j} \| \tilde{c}_{ij}^T - \tilde{G}_{ij} \tilde{c}_{ji}^T \|, \quad \text{s.t. } \tilde{G}_{ii} = I_2.$$ (3.18)

This is a SDR for the LUD problem (3.7). The problem (3.18) can be solved using ADMM (see details in section 3.5.2).

### 3.3.4 The Randomized Rounding Procedure

The matrix $R$ is recovered from a random projection of the solution $\tilde{G}$ of the SDP (3.14). We randomly draw a $2K \times 3$ matrix $P$ from the Stiefel manifold $V_3(\mathbb{R}^{2K})$. The random matrix $P$ is computed using the orthogonal matrix $Q$ and the upper triangular matrix $R$ from QR factorization of a random matrix with standard i.i.d Gaussian entries, that is, $P = Q \text{ sign} (\text{diag} (R))$, where sign stands for the entry-wise
sign function and $\text{diag}(R)$ is a diagonal matrix whose diagonal entries are the same as those of the matrix $R$. The matrix $P$ is shown to be drawn uniformly from the Stiefel manifold in [55]. We project the solution $\tilde{G}$ onto the subspace spanned by the three columns of the matrix $\tilde{GP}$.

The $2K \times 3$ matrix $\tilde{GP}$ is a proxy to the matrix $\tilde{R}^T$ (up to a global $3 \times 3$ orthogonal transformation). In other words, we can regard the $3 \times 2K$ matrix $(\tilde{GP})^T$ as composed from $K$ matrices of size $3 \times 2$, denoted $A_i$ ($i = 1, \ldots, K$), namely,

$$(\tilde{GP})^T = \begin{pmatrix} A_1 & A_2 & \cdots & A_K \end{pmatrix}$$

The two columns of each $A_i$ correspond to $R_{1i}$ and $R_{2i}$ (compare to (3.10)). We therefore estimate the matrix $R_{[1,2]}^i = \begin{pmatrix} R_{1i} & R_{2i} \end{pmatrix}$ as the closest matrix to $A_i$ on the Stiefel manifold $V_2(\mathbb{R}^3)$ in the Frobenius matrix norm. The closest matrix is given by (see, e.g., [4]) $R_{[1,2]}^i = U_i V_i^T$, where $A_i = U_i \Sigma_i V_i^T$ is the singular value decomposition of $A_i$. We note that except for the orthogonality constraint (3.15), the semidefinite program (3.14)–(3.16) is identical to the Goemans–Williamson SDP for finding the maximum cut in a weighted graph [30], where the SDR and the randomized rounding procedure [89, 50] for maximum cut problem is proved to have a 0.87 performance guarantee. From the complexity point of view, SDP can be solved in polynomial time to any given precision. The idea of using SDP for determining image orientations in cryo-EM was originally proposed in [86].

1The 3 dimensional subspace can also be spanned by the eigenvectors associated with the top three eigenvalues of $\tilde{G}$, while the fourth largest eigenvalue is expected to be significantly smaller; see also [86].
3.3.5 Exact Recovery of the Gram matrix $\tilde{G}$ from Correct Common-lines

Here we prove that if the detected common-lines $\tilde{c}_{ji}$ (defined in (3.2)) are all correct and at least three images have linearly independent projection directions (i.e., the viewing directions of the three images are not on the same great circle on the sphere shown in Figure 3.2), then the Gram matrix $\tilde{G}$ obtained by solving the LS problem (3.14)-(3.16) or the LUD problem (3.18) is uniquely the one defined in (3.11). To verify the uniqueness of the solution $\tilde{G}$, it is enough to show $\text{rank}(\tilde{G}) = 3$ due to the SDP solution uniqueness theorem (page 36-39 in [17], [119]). Without loss of generality, we consider the SDP for the LS approach when applied on three images (i.e., $K = 3$ and $w_{ij} = 1$ in the problem (3.14) - (3.16)):

$$\max_{\tilde{G}_{6 \times 6} \succeq 0} \sum_{i,j=1,2,3} \langle \tilde{G}_{ij}, \tilde{c}_{ji}^T \tilde{c}_{ij} \rangle \quad \text{s.t. } \tilde{G}_{ii} = I_2,$$

Since the solution $\tilde{G}$ is positive semidefinite, we can decompose $\tilde{G}$ as

$$\tilde{G} = \begin{pmatrix} u_1^T \\ u_1^T \\ u_2^T \\ u_2^T \\ u_3^T \\ u_3^T \end{pmatrix} \begin{pmatrix} u_1^1 & u_1^2 & u_2^1 & u_2^2 & u_3^1 & u_3^2 \end{pmatrix},$$

where $u_i^p$, $p = 1, 2$ and $i = 1, 2, 3$ are column vectors. We will show $\text{rank}(\tilde{G}) = 3$, i.e., any four vectors among $\{u_1^1, u_1^2, u_2^1, u_2^2, u_3^1, u_3^2\}$ span a space with dimensionality at most 3.
Define arrays $\mathbf{u}_i$ as
\[ \mathbf{u}_i = (\mathbf{u}^1_i, \mathbf{u}^2_i), \]
then the inner product
\[
\langle \tilde{G}_{ij}, \tilde{c}_{ji}^T \tilde{c}_{ij} \rangle = \langle \mathbf{u}_i^T \mathbf{u}_j, \tilde{c}_{ji}^T \tilde{c}_{ij} \rangle = \langle \tilde{c}_{ji} \mathbf{u}_i, \tilde{c}_{ij} \mathbf{u}_j \rangle = \langle \tilde{c}_{ji}^1 \mathbf{u}^1_i + \tilde{c}_{ji}^2 \mathbf{u}^2_i, \tilde{c}_{ij}^1 \mathbf{u}^1_j + \tilde{c}_{ij}^2 \mathbf{u}^2_j \rangle \leq 1,
\]
where the last inequality follows the Cauchy-Schwarz inequality and the facts that all $\tilde{c}_{ij}$ are unit vectors, $\mathbf{u}^1_i$ and $\mathbf{u}^2_i$ are unit vectors and orthogonal to each other due to the constraint $\tilde{G}_{ii} = I_2$, and thus all $\tilde{c}_{ji}^1 \mathbf{u}^1_j + \tilde{c}_{ij}^2 \mathbf{u}^2_j$ are unit vectors on the Fourier slices of the images. The equality holds if and only if
\[ \tilde{c}_{ji}^1 \mathbf{u}^1_i + \tilde{c}_{ji}^2 \mathbf{u}^2_i = \tilde{c}_{ij}^1 \mathbf{u}^1_j + \tilde{c}_{ij}^2 \mathbf{u}^2_j. \tag{3.19} \]
Thus when the maximum is achieved, due to (3.19) and the fact that the projection directions of the images are linearly independent, $\dim(\text{span}\{\mathbf{u}^1_i, \mathbf{u}^2_i\} \cap \text{span}\{\mathbf{u}^1_j, \mathbf{u}^2_j\}) = 1$ and thus $\dim(\text{span}\{\mathbf{u}^1_i, \mathbf{u}^2_i, \mathbf{u}^1_j, \mathbf{u}^2_j\}) = 3$. Therefore, without loss of generality, we only have to show that $\dim(\text{span}\{\mathbf{u}^1_i, \mathbf{u}^2_i, \mathbf{u}^3_i, \mathbf{u}^3_j\}) \leq 3$. Using (3.19), assume that $\text{span}\{\mathbf{u}^1_i, \mathbf{u}^2_i\} \cap \text{span}\{\mathbf{u}^1_j, \mathbf{u}^2_j\} = \text{span}\{\mathbf{v}_1\}$ and $\text{span}\{\mathbf{u}^3_i, \mathbf{u}^3_j\} \cap \text{span}\{\mathbf{v}^1_i, \mathbf{v}^3_j\} = \text{span}\{\mathbf{v}_2\}$, where $\mathbf{v}_1$ and $\mathbf{v}_2$ are linearly independent vectors (otherwise all three projection directions are linearly dependent and thus the 3 Fourier slices of the images intersect at the same line). Therefore we have $\text{span}\{\mathbf{v}_1, \mathbf{v}_2\} = \text{span}\{\mathbf{u}^1_i, \mathbf{u}^2_i\}$, $\text{span}\{\mathbf{v}_1, \mathbf{u}^1_i\} \subseteq \text{span}\{\mathbf{u}^1_i, \mathbf{u}^2_i\}$ and $\text{span}\{\mathbf{v}_2, \mathbf{u}^1_i\} \subseteq \text{span}\{\mathbf{u}^2_i, \mathbf{u}^2_i\}$. Thus $\dim(\text{span}\{\mathbf{u}^1_i, \mathbf{u}^2_i, \mathbf{u}^3_i, \mathbf{u}^3_j\}) = \dim(\text{span}\{\mathbf{u}^1_i, \mathbf{u}^2_i, \mathbf{v}_1, \mathbf{v}_2\}) \leq \dim(\text{span}\{\mathbf{u}^1_i, \mathbf{u}^2_i, \mathbf{u}^1_i, \mathbf{u}^2_i\}) \leq 3$. 

101
3.4 The Spectral Norm Constraint

In our numerical experiments (see Section 3.7), we observed that in the presence of many “outliers” (i.e., a large proportion of misidentified common-lines), the estimated viewing directions\(^2\) that are obtained by either solving (3.14)-(3.16) or (3.18) are highly clustered (Figure 3.2). This empirical behavior of the solutions can be explained by the fact that images whose viewing directions are parallel share many common lines. In other words, when the viewing directions of \(R_i\) and \(R_j\) are nearby, the fidelity term \(\left\| R_i (\vec{c}_{ij}, 0)^T - R_j (\vec{c}_{ji}, 0)^T \right\|\) (that appears in all cost functions) can become small (i.e., close to 0), even when the common line pair \((\vec{c}_{ij}, \vec{c}_{ji})\) is misidentified.

In order to prevent the viewing directions from clustering, we add the following constraint on the spectral norm of the Gram matrix \(\tilde{G}\) to the optimization problem (3.14)-(3.16) or (3.18):

\[
\tilde{G} \preceq \alpha K I_{2K},
\]

\(^2\)The viewing direction is the third column of the underlying rotation matrix.
where $I_{2K}$ is the $2K \times 2K$ identity matrix, or equivalently

$$
\left\| \tilde{G} \right\|_2 \leq \alpha K,
$$

where $\left\| \tilde{G} \right\|_2$ is the spectral norm of the matrix $\tilde{G}$, and the parameter $\alpha \in \left[ \frac{2}{3}, 1 \right)$ controls the spread of the viewing directions. If the true image orientations are uniformly sampled from the rotation group $\text{SO}(3)$, then by the law of large numbers and the symmetry of the distribution of orientations, the spectral norm of the true Gram matrix $\tilde{G}_{\text{true}}$ is approximately $\frac{2}{3}K$ (To see this, notice that $\text{Tr}(\tilde{G}) = \text{Tr}(R^T R) = \text{Tr}(RR^T) = \text{Tr}(KI_2) = 2K$. Thus, the sum of eigenvalues of $\tilde{G}$ is $2K$. Recall that $\tilde{G}$ is of rank 3, so if the rotations are uniformly distributed then each of its three non-trivial eigenvalues equals $\frac{2K}{3}$). On the other hand, if the true viewing directions are highly clustered, then the spectral norm of the true Gram matrix $\tilde{G}_{\text{true}}$ is close to $K$. For a known distribution of orientations, we can compute the spectral norm of the true Gram matrix $\tilde{G}_{\text{true}}$ accordingly, which can be verified to be a number between $\frac{2}{3}$ and 1. In practice, however, the distribution of the viewing directions is usually unknown a-priori, and often it cannot be assumed to be uniform. To prevent a solution with clustered viewing directions, we fix the parameter $\alpha$ to some number satisfying $\frac{2}{3} \leq \alpha < 1$, and perhaps even try a few possible values for $\alpha$ and choose the best value by examining the resulting reconstructions.

### 3.5 The Alternating Direction Method of Multipliers (ADMM) for SDRs with Spectral Norm Constraint

The application of ADMM to SDP problems was considered in [109]. Here we generalize the application of ADMM to the optimization problems considered in previous
sections. ADMM is a multiple-splitting algorithm that minimizes the augmented La-
grangian function in an alternating fashion such that in each step it minimizes over
one block of the variables with all other blocks fixed, and then update the Lagrange
multipliers. We apply ADMM to the dual problems since the linear constraints (3.23)
satisfy $\mathcal{A}\mathcal{A}^\ast = I$ which simplifies the computation of subproblems. The strong duality
theorem, which is known as Slater’s theorem, guarantees that in the presence of a
strictly feasible solution, a primal problem can be solved by solving its dual prob-
lem. To obtain a strictly feasible solution to the primal problems with the positive
semidefinite constraint, the linear constraint (3.23) and the spectral norm constraint
(3.24), we can construct a Gram matrix $\tilde{G}$ in (3.11) using rotations sampled from a
uniform distribution over the rotation group. Therefore, strong duality holds for the
primal problems, and the primal problems can be solved by applying ADMM to their
corresponding dual problems.

3.5.1 The Relaxed Weighted LS Problem

The weighted LS problem after SDR (3.14)-(3.16) can be efficiently solved using
SDPLR [11]. However, SDPLR is not suitable for the problem after the spectral
norm constraint on $\tilde{G}$ (3.21) is added to (3.14)-(3.16). This is because the constraint
(3.21) can be written as $\alpha K I - \tilde{G} \succeq 0$, but $\alpha K I - \tilde{G}$ does not have a low rank
structure. Moreover, SDP solvers using polynomial-time primal-dual interior point
methods are designed for small to medium sized problems. Therefore, they are not
suitable for our problem. Instead, we devise here a version of ADMM which takes
advantage of the low-rank property of $\tilde{G}$. After the spectral norm constraint (3.21)
is added, the problem (3.14)-(3.16) becomes

$$\begin{align*}
\min_{\tilde{G} \succ 0} & - \langle C, \tilde{G} \rangle \\
\text{s.t.} & \quad \mathcal{A} (\tilde{G}) = b \\
& \quad \|\tilde{G}\|_2 \leq \alpha K
\end{align*}$$

(3.22)

where

$$\mathcal{A} (\tilde{G}) = \begin{pmatrix}
\tilde{G}_{ii}^{11} \\
\tilde{G}_{ii}^{22} \\
\frac{\sqrt{2}}{2} \tilde{G}_{ii}^{12} + \frac{\sqrt{2}}{2} \tilde{G}_{ii}^{21}
\end{pmatrix}_{i=1,2,\ldots,K}, \quad b = \begin{pmatrix}
b_1^1 \\
b_2^2 \\
b_3^3
\end{pmatrix}_{i=1,2,\ldots,K}$$

(3.25)

$$b_1^1 = b_2^2 = 1, \quad b_3^3 = 0$$

for all $i$.

$\tilde{G}_{ij}^{pq}$ denotes the $(p, q)$ th element in the $2 \times 2$ sub-block $\tilde{G}_{ij}$, $C = W \circ S$ is a symmetric matrix and $\langle C, \tilde{G} \rangle = \text{trace}(C\tilde{G})$. Following the equality $\langle \mathcal{A} (\tilde{G}), y \rangle = \langle \tilde{G}, \mathcal{A}^* (y) \rangle$ for arbitrary $y = \begin{pmatrix}
y_1^1 \\
y_2^2 \\
y_3^3
\end{pmatrix}_{i=1,2,\ldots,K}$, the adjoint of the operator $\mathcal{A}$ is defined as

$$\mathcal{A}^* (y) = Y = \begin{pmatrix}
Y_{ii}^{11} & Y_{ii}^{12} \\
Y_{ii}^{21} & Y_{ii}^{22}
\end{pmatrix},$$

where for $i = 1, 2, \ldots, K$

$$Y_{ii}^{11} = y_1^1, \quad Y_{ii}^{22} = y_2^2, \quad \text{and} \quad Y_{ii}^{12} = Y_{ii}^{21} = y_3^3/\sqrt{2}.$$

It can be verified that $\mathcal{A} \mathcal{A}^* = I$. The dual problem of problem (3.22)-(3.24) is

$$\max_{y, X \succ 0, \|\tilde{G}\|_2 \leq \alpha K} \min \ - \langle C, \tilde{G} \rangle - \langle y, \mathcal{A} (\tilde{G}) - b \rangle - \langle \tilde{G}, X \rangle.$$  

(3.26)
By rearranging terms in (3.26), we obtain

$$
\max_{y,X \succeq 0 \|\tilde{G}\|_2 \leq \alpha K} \min_{\|\tilde{g}\|_2} - \left\langle C + X + \mathcal{A}^* (y), \tilde{G} \right\rangle + y^T b.
$$

(3.27)

Using the fact that the dual norm of the spectral norm is the nuclear norm (Proposition 2.1 in [76]), we can obtain from (3.27) the dual problem

$$
\max_{y,X \succeq 0} y^T b - \alpha K \|C + X + \mathcal{A}^* (y)\|_*,
$$

(3.28)

where $\|\cdot\|_*$ denotes the nuclear norm. Introducing a variable $Z = C + X + \mathcal{A}^* (y)$, we obtain from (3.28) that

$$
\min_{y,X \succeq 0} -y^T b + \alpha K \|Z\|_*
$$

s.t. $Z = C + X + \mathcal{A}^* (y)$.

(3.29)

(3.30)

Since $Z$ is a symmetric matrix, $\|Z\|_*$ is the summation of the absolute values of the eigenvalues of $Z$. The augmented Lagrangian function of (3.29)-(3.30) is defined as

$$
\mathcal{L} \left( y, Z, X, \tilde{G} \right) = -y^T b + \alpha K \|Z\|_* + \left\langle \tilde{G}, C + X + \mathcal{A}^* (y) - Z \right\rangle + \frac{\mu}{2} \|C + X + \mathcal{A}^* (y) - Z\|^2_F,
$$

(3.31)

where $\mu > 0$ is a penalty parameter. Using the augmented Lagrangian function (3.31), we devise an ADMM that minimizes (3.31) with respect to $y$, $Z$, $X$, and $\tilde{G}$ in an alternating fashion, that is, given some initial guess, in each iteration the following
three subproblems are solved sequentially:

\[
\begin{align*}
    y^{k+1} &= \arg \min_y \mathcal{L}(y, Z^k, X^k, \tilde{G}^k), \quad (3.32) \\
    Z^{k+1} &= \arg \min_Z \mathcal{L}(y^{k+1}, Z, X^k, \tilde{G}^k), \quad (3.33) \\
    X^{k+1} &= \arg \min_{X \succeq 0} \mathcal{L}(y^{k+1}, Z^{k+1}, X, \tilde{G}^k), \quad (3.34)
\end{align*}
\]

and the Lagrange multiplier \( \tilde{G} \) is updated by

\[
\tilde{G}^{k+1} = \tilde{G}^k + \gamma \mu \left( C + X^{k+1} + \mathcal{A}^* (y^{k+1}) - Z^{k+1} \right), \quad (3.35)
\]

where \( \gamma \in \left( 0, \frac{1 + \sqrt{5}}{2} \right) \) is an appropriately chosen step length.

To solve the subproblem \((3.32)\), we use the first order optimality condition

\[
\nabla_y \mathcal{L}(y, Z^k, X^k, \tilde{G}^k) = 0
\]

and the fact that \( \mathcal{A} \mathcal{A}^* = I \), and we obtain

\[
y^{k+1} = -\mathcal{A} \left( C + X^k - Z^k \right) - \frac{1}{\mu} \left( \mathcal{A} \left( \tilde{G} \right) - b \right).
\]

By rearranging the terms of \( \mathcal{L}(y^{k+1}, Z^k, X^k, \tilde{G}^k) \), it can be verified that the subproblem \((3.33)\) is equivalent to

\[
\min_Z \frac{\alpha K}{\mu} \| Z \|_* + \frac{1}{2} \| Z - B^k \|_F^2,
\]

where \( B^k = C + X^k + \mathcal{A}^* (y^{k+1}) + \frac{1}{\mu} \tilde{G}^k \). Let \( B^k = U \Lambda U^T \) be the spectral decomposition of the matrix \( B^k \), where \( \Lambda = \text{diag} (\lambda) = \text{diag} (\lambda_1, \ldots, \lambda_{2K}) \). Then \( Z^{k+1} = U \text{diag} (\tilde{\lambda}) U^T \), where \( \tilde{\lambda} \) is the optimal solution of the problem

\[
\min_z \frac{\alpha K}{\mu} \| z \|_1 + \frac{1}{2} \| z - \lambda \|_2^2,
\]  

(3.36)
It can be shown that the unique solution of (3.36) admits a closed form called the soft-thresholding operator, following a terminology introduced by Donoho and Johnstone [20]; it can be written as

\[
\hat{z}_i = \begin{cases} 
0, & \text{if } |\lambda_i| \leq \alpha K/\mu \\
(1 - \frac{\alpha}{\mu} |\lambda_i|) \lambda_i, & \text{otherwise.}
\end{cases}
\]

The problem (3.34) can be shown to be equivalent to

\[
\min_X \| X - H^k \|_F^2, \text{ s.t. } X \succeq 0,
\]

where \( H^k = Z^{k+1} - C - A^* (y^{k+1}) - \frac{1}{\mu} \tilde{G}^k \). The solution \( X^{k+1} = V_+ \Sigma_+ V_+^T \) is the Euclidean projection of \( H^k \) onto the semidefinite cone (section 8.1.1 in [8]), where

\[
V \Sigma V^T = \begin{pmatrix} V_+ & V_- \end{pmatrix} \begin{pmatrix} \Sigma_+ & 0 \\
0 & \Sigma_- \end{pmatrix} \begin{pmatrix} V_+^T \\
V_-^T \end{pmatrix}
\]

is the spectral decomposition of the matrix \( H^k \), and \( \Sigma_+ \) and \( \Sigma_- \) are the positive and negative eigenvalues of \( H^k \).

It follows from the update rule (3.35) that

\[
\tilde{G}^{k+1} = (1 - \gamma) \tilde{G}^k + \gamma \mu \left( C + X^{k+1} + A^* (y^{k+1}) - Z^{k+1} + \frac{1}{\mu} \tilde{G}^k \right)
\]

\[
= (1 - \gamma) \tilde{G}^k + \gamma \mu (X^{k+1} - H^k).
\]

### 3.5.2 The Relaxed LUD problem

Consider the LUD problem after SDR:

\[
\min_{G \succeq 0} \sum_{i<j} \| \tilde{c}_{ij}^p - \tilde{G}_{ij} \tilde{c}_{ij}^p \| \text{ s.t. } A(\tilde{G}) = b,
\]

(3.37)
where $\tilde{G}$, $A$ and $b$ are defined in (3.11) and (3.25) respectively. The ADMM devised to solve (3.37) is similar to and simpler than the ADMM devised to solve the one with the spectral norm constraint. We focus on the more difficult problem with the spectral norm constraint. Introducing $x_{ij} = \vec{c}_i^T - \tilde{G}_{ij}\vec{c}_j^T$ and adding the spectral norm constraint $\|\tilde{G}\|_2 \leq \alpha K$, we obtain

$$
\min_{x_{ij}, \tilde{G} \succeq 0} \sum_{i<j} \|x_{ij}\| \text{ s.t. } A(\tilde{G}) = b, \ x_{ij} = \vec{c}_i^T - \tilde{G}_{ij}\vec{c}_j^T, \ \|\tilde{G}\|_2 \leq \alpha K. \tag{3.38}
$$

The dual problem of problem (3.38) is

$$
\max_{\theta_{ij}, y, X \succeq 0} \min_{x_{ij}, \|\tilde{G}\|_2 \leq \alpha K} \sum_{i<j} \left( \|x_{ij}\| - \langle \theta_{ij}, x_{ij} - \vec{c}_i^T + \tilde{G}_{ij}\vec{c}_j^T \rangle \right) - \langle y, A(\tilde{G}) - b \rangle - \langle \tilde{G}, X \rangle. \tag{3.39}
$$

By rearranging terms in (3.39), we obtain

$$
\max_{\theta_{ij}, y, X \succeq 0} \min_{x_{ij}, \|\tilde{G}\|_2 \leq \alpha K} - \left\langle Q(\theta) + X + A^*(y), \tilde{G} \right\rangle + y^T b + \sum_{i<j} \left( \|x_{ij}\| - \langle \theta_{ij}, x_{ij} \rangle + \langle \theta_{ij}, \vec{c}_i^T \rangle \right), \tag{3.40}
$$

where $\theta = (\theta_{ij})_{i,j=1,...,K}$, $\theta_{ij} = (\theta_{ij}^1, \theta_{ij}^2)^T$, $\vec{c}_{ij} = (c_{ij}^1, c_{ij}^2)$,

$$
Q(\theta) = \frac{1}{2} \begin{pmatrix} Q^{11}(\theta) & Q^{12}(\theta) \\ Q^{21}(\theta) & Q^{22}(\theta) \end{pmatrix} \quad \text{and} \quad Q^{pq}(\theta) = \begin{pmatrix} 0 & \theta_{12}^p c_{21}^q & \cdots & \theta_{1K}^p c_{K1}^q \\ c_{21}^q \theta_{12}^p & 0 & \cdots & \theta_{2K}^p c_{K2}^q \\ \vdots & \vdots & \ddots & \vdots \\ c_{K1}^q \theta_{1K}^p & c_{K2}^q \theta_{2K}^p & \cdots & 0 \end{pmatrix}
$$

for $p, q = 1, 2$. It is easy to verify that for $1 \leq i < j \leq K$

$$
\min_{x_{ij}} (\|x_{ij}\| - \langle \theta_{ij}, x_{ij} \rangle) = \begin{cases} 0 & \text{if } \|\theta_{ij}\| \leq 1 \\ -\infty & \text{otherwise.} \end{cases} \tag{3.41}
$$
In fact, (3.41) is obtained using the inequality
\[
\|x_{ij}\| - \langle \theta_{ij}, x_{ij} \rangle = \|x_{ij}\| - \|\theta_{ij}\| \|x_{ij}\| \langle \theta_{ij}/\|\theta_{ij}\|, x_{ij}/\|x_{ij}\| \rangle \\
\geq \|x_{ij}\| - \|\theta_{ij}\| \|x_{ij}\| = (1 - \|\theta_{ij}\|) \|x_{ij}\|,
\] (3.42)
and the inequality in (3.42) holds when \(\theta_{ij}\) and \(x_{ij}\) have the same direction. Using the fact that the dual norm of the spectral norm is the nuclear norm and the fact in (3.41), we can obtain from (3.40) the dual problem
\[
\min_{\theta_{ij}, y, X, \geq 0} -y^T b - \sum_{i<j} \langle \theta_{ij}, \tilde{c}_{ij}^T \rangle + \alpha K \|Z\|_*
\text{s.t. } \quad Z = Q(\theta) + X + A^*(y), \quad \text{and } \|\theta_{ij}\| \leq 1.
\] (3.43)

The augmented Lagrangian function of problem (3.43)-(3.44) is defined as
\[
\mathcal{L}(y, \theta, Z, X, G) = -y^T b + \alpha K \|Z\|_* - \sum_{i<j} \langle \theta_{ij}, \tilde{c}_{ij} \rangle + \langle \tilde{G}, Q(\theta) + X + A^*(y) - Z \rangle \\
+ \mu \|Q(\theta) + X + A^*(y) - Z\|_F^2,
\] (3.45)
for \(\|\theta_{ij}\| \leq 1\), where \(\mu > 0\) is a penalty parameter. Similar to section 3.5.1, using the augmented Lagrangian function (3.45), ADMM is used to minimize (3.45) with respect to \(y, \theta, Z, X, \) and \(G\) alternatively, that is, given some initial guess, in each iteration the following four subproblems are solved sequentially:
\[
y^{k+1} = \arg \min_y \mathcal{L}\left(y, \theta^{k}, Z^k, X^k, \tilde{G}^k\right),
\] (3.46)
\[
\theta_{ij}^{k+1} = \arg \min_{\|\theta_{ij}\| \leq 1} \mathcal{L}\left(y^{k+1}, \theta, Z^k, X^k, \tilde{G}^k\right),
\] (3.47)
\[
Z^{k+1} = \arg \min_Z \mathcal{L}\left(y^{k+1}, \theta^{k+1}, Z, X^k, \tilde{G}^k\right),
\] (3.48)
\[
X^{k+1} = \arg \min_{X \geq 0} \mathcal{L}\left(y^{k+1}, \theta^{k+1}, Z^{k+1}, X, \tilde{G}^k\right),
\] (3.49)
and the Lagrange multiplier $\tilde{G}$ is updated by

$$
\tilde{G}^{k+1} = \tilde{G}^k + \gamma \mu \left( Q(\theta^{k+1}) + X^{k+1} + A^*(y^{k+1}) - Z^{k+1} \right),
$$

(3.50)

where $\gamma \in \left(0, \frac{1+\sqrt{5}}{2}\right)$ is an appropriately chosen step length. The methods to solve subproblems (3.46), (3.48) and (3.49) are similar to those used in (3.32), (3.33) and (3.34). To solve subproblem (3.47), we rearrange the terms of $L(y^{k+1}, \theta, Z^k, X^k, \tilde{G}^k)$ and obtain an equivalent problem

$$
\min_{\theta_{ij}} - \langle \theta_{ij}, \tilde{c}_{ij}^T \rangle + \frac{\mu}{2} \| \theta_{ij} \|_F^2 + \sum_{i,j} \Phi_{ij}^T (\theta_{ij}^T - \tilde{c}_{ij}^T), \quad \text{s.t. } \| \theta_{ij} \| \leq 1,
$$

where $\Phi = X + A^*(y^{k+1}) - Z + \frac{1}{\mu} \tilde{G}^k$, $\Phi = \begin{pmatrix} \Phi_{11} & \Phi_{12} \\ \Phi_{21} & \Phi_{22} \end{pmatrix}$ and $\Phi_{ij} = \begin{pmatrix} \Phi_{ij}^{11} & \Phi_{ij}^{12} \\ \Phi_{ij}^{21} & \Phi_{ij}^{22} \end{pmatrix}$.

Problem (3.50) is further simplified as

$$
\min_{\theta_{ij}} \langle \theta_{ij}, \mu \Phi_{ij} \tilde{c}_{ij}^T - \tilde{c}_{ij}^T \rangle + \frac{\mu}{2} \| \theta_{ij} \|_2^2, \quad \text{s.t. } \| \theta_{ij} \| \leq 1,
$$

whose solution is

$$
\theta_{ij} = \begin{cases} \\
\frac{1}{\mu} \tilde{c}_{ij}^T - \Phi_{ij} \tilde{c}_{ij}^T & \text{if } \| \frac{1}{\mu} \tilde{c}_{ij}^T - \Phi_{ij} \tilde{c}_{ij}^T \| \leq 1, \\
\frac{\| \tilde{c}_{ij}^T - \mu \Phi_{ij} \tilde{c}_{ij}^T \|}{\| \tilde{c}_{ij}^T - \mu \Phi_{ij} \tilde{c}_{ij}^T \|} & \text{otherwise.}
\end{cases}
$$

The practical issues related to how to take advantage of low-rank assumption of $\tilde{G}$ in the eigenvalue decomposition performed at each iteration, strategies for adjusting the penalty parameter $\mu$, the use of a step size $\gamma$ for updating the primal variable $X$ and termination rules using the in-feasibility measures are discussed in details in [109]. The convergence analysis on ADMM using more than two blocks of variables can be found in [42]. However, there is one condition of Assumption A (page 5) in [42]
that cannot be satisfied for our problem: the condition that the feasible set should be polyhedral, whereas the SDP cone in our problem is not a polyhedral. To generalize the convergence analysis in [42] to our problem, we will need to show that the local error bounds (page 8-9 in [42]) hold for the SDP cone. Currently we do not have a rigorous convergence proof for ADMM for our problem.

3.6 The Iterative Reweighted Least Squares (IRLS) Procedure

Since \( \vec{c}_{ij} \) and \( \vec{c}_{ji} \) are unit vectors, it is tempting to replace the LUD problem (3.8) with the following semidefinite relaxation:

\[
\min_{\tilde{G} \in \mathbb{R}^{2K \times 2K}} \quad F(\tilde{G}) = \sum_{i,j=1,2,\ldots,K} (2 - 2 \sum_{p,q=1,2} \tilde{G}^{pq}_{ij} S^{pq}_{ij})
\]

s.t. 
\[
\tilde{G}_{ii} = I_2, \quad i = 1,2,\ldots,K,
\]
\[
\tilde{G} \succeq 0,
\]
\[
\|\tilde{G}\|_2 \leq \alpha K \text{ (optional),}
\]

where \( \alpha \) is a fixed number between \( \frac{2}{3} \) and 1, and the spectral norm constraint on \( \tilde{G} \) (3.54) is added when the solution to the problem (3.51)-(3.53) is a set of highly clustered rotations. Notice that this relaxed problem is, however, not convex since the objective function (3.51) is concave. We propose to solve (3.51)-(3.53) (possibly with (3.54)) by an variant of the IRLS procedure [16, 47, 12], which at best converges to a local minimizer. With a good initial guess for \( \tilde{G} \) it can be hoped that the global minimizer is obtained. Such an initial guess can be taken as the LS solution.

Before the rounding procedure, the IRLS procedure finds an approximate solution to the optimization problem (3.51)-(3.53) (possibly with (3.54)) by solving its
Algorithm 1 (the IRLS procedure) Solve optimization problem (3.51)-(3.53) (with the spectral norm constraint on $\tilde{G}$ (3.54) if the input parameter $\alpha$ satisfies $\frac{2}{3} \leq \alpha < 1$), and then recover the orientations by rounding.

**Require:** a $2K \times 2K$ common-line matrix $S$, a regularization parameter $\epsilon$, a parameter $\alpha$ and the total number of iterations $N_{\text{iter}}$

$$w_{ij}^0 = 1 \ \forall i, j = 1, \ldots, K;$$

$\tilde{G}^0 = 0$;

for $k = 1 \rightarrow N_{\text{iter}}$, step size = 1 do

update $W$ by setting $w_{ij}^k = w_{ij}^{k-1}$;

if $\frac{2}{3} \leq \alpha < 1$, obtain $\tilde{G}^k$ by solving the problem (3.22)-(3.24) using ADMM; otherwise, obtain $\tilde{G}^k$ by solving (3.14)-(3.16) using SDPLR (with initial guess $\tilde{G}^{k-1}$);

$$r_{ij}^k = \sqrt{2 - 2 \sum_{p,q=1}^{2} \tilde{G}_{pq}^p S_{ij}^p + \epsilon^2};$$

$$w_{ij}^k = 1/r_{ij}^k;$$

the residual $r^k = \sum_{i,j=1}^{K} r_{ij}^k$;

end for

obtain estimated orientations $\hat{R}_1, \ldots, \hat{R}_K$ from $\tilde{G}^{N_{\text{iter}}}$ using the randomized rounding procedure in section 3.3.4.

Smoothing version

$$\min_{\tilde{G} \in \mathbb{R}^{2K \times 2K}} F(\tilde{G}, \epsilon) = \sum_{i,j=1,2,\ldots,K} \sqrt{2 - 2 \sum_{p,q=1}^{2} \tilde{G}_{ij}^p S_{ij}^p + \epsilon^2}$$

s.t. $\tilde{G}_{ii} = I_2, \ i = 1, 2, \ldots, K,$

$\tilde{G} \succ 0,$

$$\|\tilde{G}\|_2 \leq \alpha K \ (\text{optional}).$$

where $\epsilon > 0$ is a small number. The solution to the smoothing version is close to the solution to the original problem. In fact, let $\tilde{G}_\epsilon^* = \arg \min F(\tilde{G}, \epsilon)$ and $\tilde{G}^* = \arg \min F(\tilde{G})$, then we shall verify that

$$\left| F(\tilde{G}_\epsilon^*) - F(\tilde{G}^*) \right| \leq 4K^2 \epsilon.$$  

(3.59)
Using the fact that
\[ 0 \leq F(\tilde{G}, \epsilon) - F(\tilde{G}) < 4K^2\epsilon, \]
we obtain
\[
(F(\tilde{G}^*, \epsilon) - F(\tilde{G}^*_\epsilon, \epsilon)) + (F(\tilde{G}^*_\epsilon, \epsilon) - F(\tilde{G}^*))
\leq 4K^2\epsilon.
\]
Since \( F(\tilde{G}^*, \epsilon) - F(\tilde{G}^*_\epsilon, \epsilon) \geq 0 \) and \( F(\tilde{G}^*_\epsilon, \epsilon) - F(\tilde{G}^*) \geq 0 \), the inequality (3.59) holds.

In each iteration, the IRLS procedure solves the problem
\[
\tilde{G}^{k+1} = \arg \min_{\tilde{G} \succ 0} \sum_{i \neq j} w_{ij}^k \left( 2 - 2 \langle \tilde{G}_{ij}, S_{ij} \rangle + \epsilon^2 \right)
\text{s.t. } \mathcal{A}(\tilde{G}) = b, \text{ (optional: } \left\| \tilde{G} \right\|_2 \leq \alpha K) \quad (3.60)
\]
on the \((k + 1)\)th iteration, where \( w_{ij}^0 = 1 \), and
\[
w_{ij}^k = 1/\sqrt{2 - 2 \langle \tilde{G}_{ij}^k, S_{ij} \rangle + \epsilon^2}, \forall k > 0.
\]
In other words, in each iteration, more emphasis is given to detected common-lines that are better explained by the current estimate \( \tilde{G}^k \) of the Gram matrix. The inclusion of the regularization parameter \( \epsilon \) ensures that no single detected common-line can gain undue influence when solving
\[
\tilde{G}^{k+1} = \arg \max_{\tilde{G} \succ 0} \left\langle W^k \circ S, \tilde{G} \right\rangle \text{ s.t. } \mathcal{A}(\tilde{G}) = b \text{ (optional: } \left\| \tilde{G} \right\|_2 \leq \alpha K). \quad (3.61)
\]
We repeat the process until the residual sequence \( \{r^k\} \) has converged, or the maximum number of iterations has been reached. We shall verify that the value of the cost function is non-increasing, and that every cluster point of the sequence of IRLS is a
stationary point of (3.55) - (3.57) in the following lemma and theorem, for the problem without the spectral norm constraint on $\tilde{G}$. The arguments can be generalized to the case with the spectral norm constraint. The proof of Theorem 3.6.2 follows the method of proof for Theorem 3 in the paper [57] by Mohan et. al..

**Lemma 3.6.1** The value of the cost function sequence is monotonically non-increasing, i.e.,

$$F(\tilde{G}^{k+1}, \epsilon) \leq F(\tilde{G}^k, \epsilon).$$

(3.62)

where $\{\tilde{G}^k\}$ is the sequence generated by the IRLS procedure of Algorithm 1.

**Proof** Since $\tilde{G}^k$ is the solution of (3.61), there exists $y^k \in \mathbb{R}^{2K}$ and $X^k \in \mathbb{R}^{2K \times 2K}$ such that

$$-A^*(y^k) + X^k + W^{k-1} \circ S = 0, \quad A(\tilde{G}^k) - b = 0,$$

$$\tilde{G}^k \succ 0, \quad X^k \succ 0, \quad \langle \tilde{G}^k, X^k \rangle = 0.$$ (3.63)

Hence we have

$$0 = -(y^k)^T (A(\tilde{G}^k) - b) + (y^{k+1})^T (A(\tilde{G}^{k+1}) - b)$$

$$= (y^{k+1} - y^k)^T (A(\tilde{G}^k) - b) + \left\langle A^*(y^{k+1}), \tilde{G}^{k+1} - \tilde{G}^k \right\rangle$$

$$= \left\langle X^{k+1} + W^k \circ S, \tilde{G}^{k+1} - \tilde{G}^k \right\rangle$$

$$\leq \left\langle W^k \circ S, \tilde{G}^{k+1} - \tilde{G}^k \right\rangle$$

$$= \frac{1}{2} \sum_{i \neq j} (\beta^k_{ij} \left( 2 - 2 \left\langle \tilde{G}_{ij}^{k+1}, S_{ij} \right\rangle + \epsilon^2 \right) + \beta^k_{ij} \left( 2 - 2 \left\langle \tilde{G}_{ij}^k, S_{ij} \right\rangle + \epsilon^2 \right))$$

$$= \frac{1}{2} \sum_{i \neq j} \left( -\frac{2 - 2 \left\langle \tilde{G}_{ij}^{k+1}, S_{ij} \right\rangle + \epsilon^2}{\sqrt{2 - 2 \left\langle \tilde{G}_{ij}^k, S_{ij} \right\rangle + \epsilon^2}} + \sqrt{2 - 2 \left\langle \tilde{G}_{ij}^k, S_{ij} \right\rangle + \epsilon^2} \right),$$ (3.66)
where the third equality uses (3.63), and the inequality (3.65) uses (3.64). From (3.66) we obtain

\[
F(\tilde{G}^k, \epsilon)^2 = \left( \sum_{i \neq j} \sqrt{2 - 2 \langle \tilde{G}^k_{ij}, S_{ij} \rangle + \epsilon^2} \right)^2 \geq \left( \sum_{i \neq j} \frac{2 - 2 \langle \tilde{G}^{k+1}_{ij}, S_{ij} \rangle + \epsilon^2}{\sqrt{2 - 2 \langle \tilde{G}^k_{ij}, S_{ij} \rangle + \epsilon^2}} \right)^2 \geq \left( \sum_{i \neq j} \sqrt{2 - 2 \langle \tilde{G}^{k+1}_{ij}, S_{ij} \rangle + \epsilon^2} \right)^2 = F(\tilde{G}^{k+1}, \epsilon)^2, \tag{3.67}
\]

where the last inequality uses Cauchy-Schwarz inequality and the equality holds if and only if

\[
\sqrt{2 - 2 \langle \tilde{G}^{k+1}_{ij}, S_{ij} \rangle + \epsilon^2} = c \text{ for all } i \neq j, \tag{3.68}
\]

where \(c\) is a constant. Thus (3.62) is confirmed.

**Theorem 3.6.2** The sequence of iterates \(\{\tilde{G}^k\}\) of IRLS is bounded, and every cluster point of the sequence is a stationary point of (3.55) - (3.57).

**Proof** Since \(\text{trace}(\tilde{G}^k) = 2K\) and \(\tilde{G}^k \succeq 0\), the sequence \(\{\tilde{G}^k\}\) is bounded. It follows that \(W^k\) and \(\text{trace}((W^k \circ S)\tilde{G}^{k+1})\) are bounded. Using the strong duality of SDP, we conclude that \(b^T y^{k+1} = \text{trace}((W^k \circ S)\tilde{G}^{k+1})\) is bounded. In addition, from the KKT conditions (3.63) - (3.64) we obtain \(-\mathcal{A}^*(y^{k+1}) + W^k \circ S \succeq 0\). Using the definition of \(\mathcal{A}^*\) and \(S\), the property of semi-definite matrices and the fact that \(W^k\) is bounded, it can be verified that \(y^k\) is bounded. Using (3.63) again, we obtain

\[
\|X^k\| = \|\mathcal{A}^*(y^k) - W^{k-1} \circ S\| \leq \|\mathcal{A}^*(y^k)\| + \|W^{k-1} \circ S\|,
\]

which implies that \(X^k\) is bounded.
We now show that every cluster point of $\{\hat{G}^k\}$ is a stationary point of (3.55) - (3.57). Suppose to the contrary and let $\bar{G}$ be a cluster point of $\{\hat{G}^k\}$ that is not a stationary point. By the definition of cluster point, there exists a subsequence $\{\hat{G}^{k_i}, \bar{W}^{k_i}, \bar{X}^{k_i}, \bar{y}^{k_i}\}$ of $\{\hat{G}^k, \bar{W}^k, \bar{X}^k, \bar{y}^k\}$ converging to $(\bar{G}, \bar{W}, \bar{X}, \bar{y})$. By passing to a further subsequence if necessary, we can assume that $\{\hat{G}^{k_i+1}, \bar{W}^{k_i+1}, \bar{X}^{k_i+1}, \bar{y}^{k_i+1}\}$ is also convergent and we denote its limit by $(\hat{G}, \hat{W}, \hat{X}, \hat{y})$. $\hat{G}^{k_i+1}$ is defined as (3.60) or (3.61) and satisfies the KKT conditions (3.63) - (3.64). Passing to limits, we see that

$$-A^*(\hat{y}) + \hat{X} + \bar{W} \circ S = 0, A(\hat{G}) - \bar{b} = 0,$$

$$\hat{G} \succ 0, \hat{X} \succ 0, \langle \hat{G}, \hat{X} \rangle = 0.$$

Thus we conclude that $\hat{G}$ is a maximizer of the following convex optimization problem,

$$\max_{\hat{G} \succeq 0} \langle \bar{W} \circ S, \hat{G} \rangle \text{ s.t. } A(\hat{G}) = \bar{b}.$$

Next, by assumption, $\bar{G}$ is not a stationary point of (3.55) - (3.57). This implies that $\bar{G}$ is not a maximizer of the problem above and thus $\langle \bar{W} \circ S, \bar{G} \rangle > \langle \bar{W} \circ S, \hat{G} \rangle$. From this last relation and (3.67) - (3.68) it follows that

$$F(\hat{G}, \epsilon) < F(\bar{G}, \epsilon). \quad (3.69)$$

Otherwise if $F(\hat{G}, \epsilon) = F(\bar{G}, \epsilon)$, then $\langle \hat{G}_{ij}, S_{ij} \rangle = \langle \bar{G}_{ij}, S_{ij} \rangle$ due to (3.67) - (3.68), and thus we would obtain $\langle \bar{W} \circ S, \hat{G} \rangle = \langle \bar{W} \circ S, \bar{G} \rangle$ which is a contradiction.
On the other hand, it follows from Lemma 3.6.1 that the sequence \( \{F(\tilde{G}^k, \epsilon)\} \) converges. Thus we have that

\[
\lim F(\tilde{G}^k, \epsilon) = \lim F(\tilde{G}^{k_i}, \epsilon) = F(\tilde{G}, \epsilon) = \lim F(\tilde{G}^{k_i+1}, \epsilon) = F(\hat{G}, \epsilon)
\]

which contradicts (3.69). Hence, every cluster point of the sequence is a stationary point of (3.55) - (3.57).

In addition, using Hölder’s inequality, the analysis can be generalized to the reweighted approach to solve

\[
\min_{\tilde{G} \succ 0} \sum_{i \neq j} \left( 2 - 2 \left\langle \tilde{G}_{ij}, S_{ij} \right\rangle \right)^{\frac{p}{2}} \text{ s.t. } A(\tilde{G}) = b, \text{ (optional: } \left\| \tilde{G} \right\|_2 \leq \alpha K) \tag{3.70}
\]

where \( 0 < p < 1 \). Convergence analysis of IRLS for different applications with \( p < 1 \) can be found in [16, 47]. The problem (3.70) is a SDR of the problem

\[
\min_{R_1, \ldots, R_K \in SO(3)} \sum_{i \neq j} \left\| R_i (\tilde{c}_{ij}, 0)^T - R_j (\tilde{c}_{ji}, 0)^T \right\|^p. \tag{3.71}
\]

The smaller \( p \) is, the more penalty the outliers in the detected common-lines receive.

### 3.7 Numerical Results

All numerical experiments were performed on a machine with 2 Intel(R) Xeon(R) CPUs X5570, each with 4 cores, running at 2.93 GHz. In all the experiments, the polar Fourier transform of images for common-line detection had radial resolution \( n_r = 100 \) and angular resolution \( n_\theta = 360 \). The number of iterations was set to be \( N_{\text{iter}} = 10 \) in all IRLS procedures. The reconstruction from the images with estimated orientations used the Fourier based 3D reconstruction package FIRM\[3]\[105]. The

\[3\]The FIRM package is available at [https://web.math.princeton.edu/~lanhuiw/software.html](https://web.math.princeton.edu/~lanhuiw/software.html).
reconstructed volumes are shown in Figure 3.4 and 3.7 using the visualization system Chimera [72].

To evaluate the accuracy or the resolution of the reconstructions, we used the 3D Fourier Shell Correlation (FSC) [80]. FSC measures the normalized cross-correlation coefficient between two 3D volumes over corresponding spherical shells in Fourier space, i.e.,

\[
\text{FSC}(i) = \frac{\sum_{j \in \text{Shell}_i} \mathcal{F}(V_1)(j) \cdot \mathcal{F}(V_2)(j)}{\sqrt{\sum_{j \in \text{Shell}_i} |\mathcal{F}(V_1)(j)|^2 \cdot \sum_{j \in \text{Shell}_i} |\mathcal{F}(V_2)(j)|^2}},
\]

where \( \mathcal{F}(V_1) \) and \( \mathcal{F}(V_2) \) are the Fourier transforms of volume \( V_1 \) and volume \( V_2 \) respectively, the spatial frequency \( i \) ranges from 1 to \( N/2 - 1 \) times the unit frequency \( 1/(N \cdot \text{pixel size}) \), \( N \) is the size of a volume, and \( \text{Shell}_i := \{ j : 0.5 + (i - 1) + \epsilon \leq \| j \| < 0.5 + i + \epsilon \} \) where \( \epsilon = 1 \times 10^{-4} \). In this form, the FSC takes two 3D volumes and converts them into a 1D array. In Section 3.7.2 we used the FSC 0.143 cutoff criterion [SI, 5] to determine the resolutions of the ab-initio models and the refined models.

### 3.7.1 Experiments on Simulated Images

We simulated 500 centered images of size 129 × 129 pixels with pixel size 2.4Å of the 50S ribosomal subunit (the top volume in Figure 3.4), where the orientations of the images were sampled from the uniform distribution over \( \text{SO}(3) \). White Gaussian noise was added to the clean images to generate noisy images with SNR= 1/16, 1/32 and 1/64 respectively (Figure 3.3). Common-line pairs that were detected with an error smaller than 10° were considered to be correct. The common-line detection rates were 64%, 44% and 23% for images with SNR=1/16, 1/32 and 1/64 respectively (Figure 3.1).
Figure 3.3: The first column shows three clean images of size 129×129 pixels generated from a 50S ribosomal subunit volume with different orientations. The other three columns show three noisy images corresponding to those in the first column with SNR= 1/16, 1/32 and 1/64, respectively.

To measure the accuracy of the estimated orientations, we defined the mean squared error (MSE) of the estimated rotation matrices $\hat{R}_1, \ldots, \hat{R}_K$ as

$$\text{MSE} = \frac{1}{K} \sum_{i=1}^{K} \| R_i - \hat{O} \hat{R}_i \|^2, \tag{3.73}$$

where $\hat{O}$ is the optimal solution to the registration problem between the two sets of rotations $\{R_1, \ldots, R_K\}$ and $\{\hat{R}_1, \ldots, \hat{R}_K\}$ in the sense of minimizing the MSE. As shown in [80], there is a simple procedure to obtain both $\hat{O}$ and the MSE from the singular value decomposition of the matrix $\frac{1}{K} \sum_{i=1}^{K} \hat{R}_i R_i^T$.

We applied the LS approach using SDP and ADMM, and the LUD approach using ADMM and IRLS to estimate the images’ orientations, then computed the MSEs of the estimated rotation matrices, and lastly reconstructed the volume (Figure 3.4). In order to measure the accuracy of the reconstructed volumes, we measured each volume’s FSC \(\text{FSC} (3.72)\) (Figure 4.2) against the clean 50S ribosomal subunit volume, that
Figure 3.4: The clean volume (top), the reconstructed volumes and the MSEs of the estimated rotations. From 2nd to 4th row, no spectral norm constraint was used (i.e., $\alpha = \text{N/A}$) for all algorithms. The last 4 rows are all results of very noisy images with SNR = 1/64, where the result using the IRLS procedure without $\alpha$ is not available due to the highly clustered estimated projection directions, and the result from the IRLS procedure with $\alpha = 0.67$ for the spectral norm constraint is best.
Figure 3.5: FSCs (3.72) of the reconstructed volumes against the clean volume in Figure 3.4. The plots of the correlations show that the LUD approach using ADMM and IRLS (denoted as the blue dashed lines and green solid lines) outweighed the LS approach using SDP or ADMM (denoted as the red dot-dashed lines). Note that all the last four sub-figures are results for images with SNR = 1/64, where the last three sub-figures are results using different \( \alpha \) for the spectral norm constraint. In the third sub-figure (left to right, top to bottom), there is no green solid line for the LUD approach using IRLS, since the IRLS procedure without the spectral norm constraint converges to a solution where the estimated projection directions are highly clustered and no 3D reconstruction can be computed.

is, in our measurement \( \mathbf{V}_1 \) was the reconstructed volume, and \( \mathbf{V}_2 \) was the “ground truth” volume.

When SNR = 1/16 and 1/32, the common-line detection rate was relatively high (64% and 44%), the algorithms without the spectral norm constraint on \( \tilde{G} \) were enough to make a good estimation. The LUD approach using ADMM and IRLS outweighed the LS approach in terms of accuracy measured by MSE and FSC (Figure 3.4-4.2). Note that the LS approach using SDP failed when SNR = 1/32, while the LUD approach using either ADMM or IRLS succeeded. When SNR=1/64, the
common-line detection rate was relatively small (23%), and most of the detected common-lines were outliers (Figure 3.1), the algorithms without spectral norm constraint $\|\tilde{G}\|_2 \leq \alpha K$ did not work. Especially, the viewing directions of images estimated by the IRLS procedures without $\|\tilde{G}\|_2 \leq \alpha K$ converged to two clusters around two antipodal directions, yielding no 3D reconstruction. The LUD approach using ADMM failed in this case, however, the IRLS procedure with an appropriate regularization on the spectral norm (i.e., $\alpha = 0.67$ since the true rotations were uniformly sampled over $SO(3)$) gave the best reconstruction.

3.7.2 Experiments on a Real Dataset

A set of micro-graphs of E. coli 50S ribosomal subunits was provided by Dr. M. van Heel. These micro-graphs were acquired by a Philips CM20 at defocus values between
Figure 3.7: Initial models and refined models. (a) The ab-initio models estimated by merging two independent reconstructions, each obtained from 1000 class averages. The resolutions of the models are 17.2Å, 16.7Å, 16.7Å, 16.7Å and 16.1Å (from top to bottom, left to right) using the FSC 0.143 resolution cutoff (Figure 3.8). The model using the IRLS procedure without the spectral norm constraint (i.e., $\alpha = N/A$) is not available since the estimated projection directions are highly clustered. (b) The refined models corresponding to the ab-initio models in (a). The resolutions of the models are all 11.1Å.
Figure 3.8: Convergence of the refinement process. In sub-figure (a) - (e), the FSC plots show the convergence of the refinement iterations. The ab-initio models (Figure 3.7a used in (a) - (e) were obtained by solving the LS/LUD problems using SDP/ADMM/IRLS. The numbers of refinement iterations performed in (a) - (e) are 7, 3, 5, 3 and 4 respectively. The sub-figure (f) are FSC plots of the refined models in (a), (b), (c) and (e) against the refined model in (d), which are measurements of similarities between the refined models in Figure 3.7b.
1.37 and 2.06 µm, and they were scanned at 3.36 Å/pixel. The particles (particularly E. coli 50S ribosomal subunits) were picked using the automated particle picking algorithm in EMAN Boxer [49]. Then using the IMAGIC software package ([93, 104]), the 27,121 particle images of size 90×90 pixels were phase-flipped to remove the phase-reversals in the CTF, bandpass filtered at 1/150 and 1/8.4 Å, normalized by their variances, and then translationally aligned with the rotationally-averaged total sum. The particle images were randomly divided into 2 disjoint groups of equal number of images. The following steps were performed to each group separately.

The images were rotationally aligned and averaged to produce class averages of better quality, following the procedure detailed in [116]. For each group, the images were denoised and compressed using Fourier-Bessel based principal component analysis (FBsPCA) [115]. Then, triple products of Fourier-Bessel expansion coefficients obtained in FBsPCA were used to compute rotational invariant features of the images, i.e, the bispectrums [78, 45, 52]. For each image, an initial set of neighboring images was computed using the normalized cross-correlation of the bispectrums, which was later refined using the method described in [88] to produce new sets of neighbors. Finally, for each image, we averaged it with its 10 nearest neighbors after alignment. Three examples of averaged images are shown in Figure 3.6.

One thousand class averages were randomly selected from each group. The LS and LUD approaches with and without the spectral norm constraint were applied. Two reconstructed volumes were obtained from the two groups of images. The two resulting volumes were aligned and averaged to obtain the ab-initio model (Figure 3.7a). We observed that the LUD approach gives much more reasonable ab initio models compared to the LS approach. In addition, the FSC of the two volumes was computed to estimate the resolution of the ab-initio model (Figure 3.8). Among all the ab-initio models, the one obtained by LS is at the lowest resolution 17.2 Å, while the one obtained by LUD through IRLS procedure is the highest resolution.
16.1Å. Notice that the FSC measures the variance error, but not the bias error of the ab-initio model. We also notice that the viewing directions of images estimated by the IRLS procedures without the spectral norm constraint converged to two clusters around two antipodal directions, resulting in no 3D reconstruction. Moreover, for this dataset, adding the spectral norm constraint on $\tilde{G}$ with $\alpha = 0.85$ did not improve the accuracy of the result, although this helped with regularizing the convergence in the IRLS procedure.

The two resulting volumes were then iteratively refined using 10,000 raw images in each group. In each refinement iteration, 2,000 template images were generated by projecting the 3D model from the previous iteration, then the orientations of the raw images were estimated using reference-template matching, and finally a new 3D model was reconstructed from the 10,000 raw images with highest correlation with the reference images. Each refinement iteration took about 4 hours. Therefore, a good ab-initio model should be able to accelerate the refinement process by reducing the total number of refinement iterations. The FSC plots in Figure 3.8a - 3.8e show the convergence of the refinement process using different ab-initio models. We observed that all the refined models are at the resolution 11.1Å. However, the worst ab-initio model obtained by LS needed 7 iterations (about 28 hours) for convergence (Figure 3.8a), while the best ab-initio model obtained by LUD needed 3 iterations (about 12 hours) for convergence (Figure 3.8b and Figure 3.8d). Figure 3.8f uses FSC plots to compare the refined models. We observed that the refined models in Figure 3.8b - 3.8e were consistent to each other, while the refined model obtained by LS in Figure 3.8a was slightly different from others.

The average cost time for computing the ab-initio models in these two subsections are shown in Table 3.1. It is not surprising to see that the LS approach was the fastest and that adding the spectral norm constraint slowed down the ADMM and IRLS procedures. The reason is that a large portion of the cost time in ADMM and
IRLS is due to the projections onto the semidefinite cone. These steps are expected to be accelerated by the recent advance on eigenspace computation [110]. However, when using the LUD approach for the real data set, the time saved in the refinement was about 16 hours, which is much more than the time cost for computing the ab-initio models (about 0.5 - 1 hour when ADMM was used).

<table>
<thead>
<tr>
<th>$K$ (SDP)</th>
<th>$\alpha = \text{N/A}$</th>
<th>$\frac{2}{3} \leq \alpha \leq 1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>LS</td>
<td>LUD</td>
<td>LS</td>
</tr>
<tr>
<td>ADMM</td>
<td>IRLS</td>
<td>ADMM</td>
</tr>
<tr>
<td>IRLS</td>
<td>LUD</td>
<td>IRLS</td>
</tr>
<tr>
<td>500</td>
<td>7s</td>
<td>469s</td>
</tr>
<tr>
<td></td>
<td>266s</td>
<td>78s</td>
</tr>
<tr>
<td>1000</td>
<td>31s</td>
<td>3913s</td>
</tr>
<tr>
<td></td>
<td>1864s</td>
<td>619s</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1928s</td>
</tr>
<tr>
<td></td>
<td></td>
<td>20918s</td>
</tr>
</tbody>
</table>

Table 3.1: The average cost time using different algorithms on 500 and 1000 images in the two experimental subsections. The notation $\alpha = \text{N/A}$ means no spectral norm constraint $\|\tilde{G}\|_2 \leq \alpha K$ is used.

### 3.8 Discussion

To estimate image orientations, we introduced a robust self consistency error and used ADMM or the IRLS procedure to solve the associated LUD problem after SDR. Numerical experiments demonstrate that the solution is less sensitive to outliers in the detected common-lines than the LS method approach. In addition, when the common-line detection rate is low, the spectral norm constraint on the Gram matrix $\tilde{G}$ can help to tighten the semidefinite relaxation, and thus improves the accuracy of the estimated rotations in some cases. Moreover, the numerical experiments using the real data set (Section 3.7.2) demonstrate that the ab-initio models resulted by the LUD based methods are more accurate than initial models that are resulted by least squares based methods. In particular, our initial models requires fewer time-consuming refinement iterations. We note that it is also possible to consider other self consistency errors involving the unsquared deviations raised to some power $p$ (e.g., the cases $p = 1, 2$ correspond to LUD and LS, respectively). We observed that the
accuracy of the estimated orientations can be improved by using $p < 1$ provided that the initial guess is “sufficiently good”. The LUD approach and the spectral norm constraint on $\tilde{G}$ can be generalized to the synchronization approach to estimate the images’ orientations in [83].

In [108], the LUD approach is shown to be more robust than the LS approach for the synchronization problem over the rotation group $SO(d)$. Given some relative rotations $R_i^T R_j$, the synchronization problem is to estimate the rotations $R_i \in SO(d)$, $i = 1, \ldots, K$ up to a global rotation. It is verified that under a specific model of the measurement noise and the measurement graph for $R_i^T R_j$, the rotations can be exactly and stably recovered using LUD, exhibiting a phase transition behavior in terms of the proportion of noisy measurements. The problem of orientation determination using common-lines between cryo-EM images is similar to the synchronization problem. The difference is that the pairwise information given by the relative rotation $R_i^T R_j$ is full, while that given by the common-lines $\vec{c}_{ji}^T \vec{c}_{ij}$ is partial. Moreover, the measurement noise of each detected common-line $\vec{c}_{ij}$ depends on image $i$ and $j$, and thus it cannot be simply modeled, which brings the difficulties in verifying the conditions for the exact and stable orientation determination we observed.
Chapter 4

A Fourier-based Approach for
Inversion from Cryo-EM Images

Obtaining high resolution 3D reconstructions by inverting a set of 2D cryo-EM images with an estimate of orientations is difficult due to possible incomplete coverage of the viewing directions, high level of noise in the projection images, and limiting effects of the contrast transfer function of the electron microscope. In this chapter, we focus on the 3D reconstruction problem from projection images assuming an existing estimate for their orientations and positions. We propose a fast and accurate Fourier-based Iterative Reconstruction Method (FIRM) that exploits the Toeplitz structure of the operator $A^*A$, where $A$ is the forward projector and $A^*$ is the back projector. The operator $A^*A$ is equivalent to a convolution with a kernel. The kernel is pre-computed using the non-uniform Fast Fourier Transform and is efficiently applied in each iteration step. The iterations by FIRM are therefore considerably faster than those of traditional iterative algebraic approaches, while maintaining the same accuracy even when the viewing directions are unevenly distributed. The time complexity of FIRM is comparable to the direct Fourier inversion method. Moreover, FIRM combines images from different defocus groups simultaneously and can handle
a wide range of regularization terms. We provide experimental results on simulated data that demonstrate the speed and accuracy of FIRM in comparison with current methods.

The outline of the chapter is as follows. In section 4.1 we introduce the necessary mathematical background concerning the reconstruction problem. In section 4.2 we demonstrate the key property that the composition of back-projection and forward-projection has a Toeplitz structure. We utilize this Toeplitz structure to accelerate the iterations in the CG method. Finally, numerical examples and concluding remarks are given in sections 4.3 and 4.4.

4.1 Preliminaries

In this section, we provide the necessary mathematical background concerning the reconstruction problem in cryo-EM and introduce notation used throughout this chapter.

4.1.1 Notation

Scalars, indices and functions are denoted by non-boldface lowercase letters such as \( x, i, \) and \( f \). Global constants are denoted by non-boldface uppercase letters such as \( C \) and \( N \). Boldface lowercase letters denote either vectors or arrays, e.g., \( \mathbf{a} = (a_{i_1,i_2,\ldots,i_d}) \) with \( i_k = 1, 2, \ldots, n_k \) and \( k = 1, 2, \ldots, d \) is a \( d \)-dimensional array of size \( n_1 \times n_2 \times \ldots \times n_d \). We refer to individual elements as either \( a_{i_1,i_2,\ldots,i_d} \) or \( a(i_1,i_2,\ldots,i_d) \). Operations such as \( \mathbf{a}/2 \) and \( \mathbf{a} > 0 \) are considered component-wise. Matrices and operators are denoted by boldface uppercase letters, such as \( \mathbf{A} \) and \( \mathbf{F} \). The elements of a matrix \( \mathbf{A} \) are denoted as either \( \mathbf{A}(i,j) \) or \( A_{i,j} \). The elements of a matrix \( \mathbf{A} \) of multi-order \( d > 1 \) are denoted as either \( \mathbf{A}(i,j) \) or \( A_{i,j} \), where \( i \) and \( j \) are vector indices. The adjoint of a matrix (or an operator) \( \mathbf{A} \) is denoted as \( \mathbf{A}^* \). The conjugate
of a complex number $z = a + ib$ is denoted as $\bar{z} = a - ib$. The absolute value of $z$ is denoted as $|z| = \sqrt{a^2 + b^2}$. The inner product of two arbitrary $n_1 \times n_2 \times \ldots \times n_d$ arrays $a$ and $b$ is defined as $\langle a, b \rangle = \sum_{i_1, i_2, \ldots, i_d} a_{i_1, i_2, \ldots, i_d} \bar{b}_{i_1, i_2, \ldots, i_d} = \sum_i a_i \bar{b}_i$. We omit the index and bounds of summation when these are clear from the context. The $\ell^2$ norm of $a$ is denoted as $\|a\| = \sqrt{\langle a, a \rangle}$.

4.1.2 Fourier Transform Conventions

The $d$-D Fourier transform $F$ of a function $f : \mathbb{R}^d \to \mathbb{C}$ is defined by

$$(Ff)(\omega) = \int_{\mathbb{R}^d} f(x) \exp(-i \langle \omega, x \rangle) dx, \quad \text{for } \omega \in \mathbb{R}^d.$$  

Likewise, the $d$-D discrete Fourier Transform (DFT) $F$ of an array $f = (f_n)$, where $-N/2 \leq n < N/2$, is given by

$$(Ff)_k = \sum_{-N/2 \leq n < N/2} f_n \exp(-i 2\pi \langle k, n \rangle / N), \quad \text{for } -N/2 \leq k < N/2.$$  

4.1.3 Toeplitz Matrices and Circulant Matrices

We will show in section 3 that reconstructing a volume is equivalent to solving a symmetric positive-definite Toeplitz system. In this subsection, we introduce Toeplitz matrices and circulant matrices. An $n \times n$ Toeplitz matrix is of the following form:

$$T_n = \begin{pmatrix} 
  t_0 & t_{-1} & \cdots & t_{2-n} & t_{1-n} \\
  t_1 & t_0 & t_{-1} & \cdots & t_{2-n} \\
  \vdots & t_1 & t_0 & \ddots & \vdots \\
  t_{n-2} & \cdots & \ddots & \ddots & t_{-1} \\
  t_{n-1} & t_{n-2} & \cdots & t_1 & t_0 
\end{pmatrix},$$

i.e., $T_n(i, j) = t_{i-j}$ and $T_n$ is constant along its diagonals.
A circulant matrix is a Toeplitz matrix of the form:

$$
C_n = \begin{pmatrix} 
  c_0 & c_{n-1} & \cdots & c_2 & c_1 \\
  c_1 & c_0 & c_{n-1} & \cdots & c_2 \\
  \vdots & c_1 & c_0 & \ddots & \vdots \\
  c_{n-2} & \cdots & \cdots & \cdots & c_{n-1} \\
  c_{n-1} & c_{n-2} & \cdots & c_1 & c_0 
\end{pmatrix},
$$

i.e., $C_n(i,j) = c_{i-j}$ where $c_{-k} = c_{n-k}$ for $1 \leq k \leq n - 1$. Note that $C_n$ is completely determined by its first column. It is well-known that circulant matrices are diagonalized by the Fourier matrix $F_n$ [94], i.e.,

$$
C_n = F_n^* \Lambda_n F_n, \tag{4.1}
$$

where $F_n(j, k) = \frac{1}{\sqrt{n}} \exp(2\pi i j k / n)$, and $\Lambda_n$ is a diagonal matrix. It follows immediately from (4.1) that the diagonal entries of $\Lambda_n$, namely, the eigenvalues of $C_n$ can be obtained in $O(n \log n)$ operations using the Fast Fourier Transform (FFT) of the first column of $C_n$. Once $\Lambda_n$ is obtained, the matrix-vector product $C_n y$ can be computed efficiently by two FFTs in $O(n \log n)$ operations using (4.1) for any vector $y$.

Similarly, we can define $n$-by-$n$ Toeplitz matrices and circulant matrices of multi-order $d$. The property (4.1) can be generalized to multi-order circulant matrices.

### 4.2 A Fourier-based Approach for 3D Inversion

#### 4.2.1 The Forward-Projector $A$

In cryo-EM, the structure of a molecule is described by the molecule’s electric potential function $\phi(x)$, where $x = (x_1, x_2, x_3)$ is in $\mathbb{R}^3$. In a cryo-EM experiment, the macromolecules are assumed to be identical with different orientations. We use $\mathbb{R}$ to
denote the rotation of each molecule, where \( \mathbf{R} \) is an element of the rotation group \( \text{SO}(3) \). The projection image of a molecule with orientation \( \mathbf{R} \) is given by

\[
(P_{\mathbf{R}}\varphi)(x_1, x_2) = \int_{-\infty}^{\infty} \varphi_{\mathbf{R}}(x_1, x_2, x_3) dx_3,
\]

where \( \varphi_{\mathbf{R}}(\mathbf{x}) = \varphi(\mathbf{R}^{-1}\mathbf{x}) \) is the electric potential of the molecule after a rotation by \( \mathbf{R} \). Note that for cryo-EM images, pose parameters include both translations and rotations. Given the translations, the images are re-shifted to their centers. Therefore, here we consider the reconstruction problem for centered images given the rotational information.

With the above definitions of the Fourier transform and the projection, one can verify the following theorem, known as the Fourier projection-slice theorem (page 11 in [59]):

\[
(FP_{\mathbf{R}}\varphi)(\omega_1, \omega_2) = (F\varphi_{\mathbf{R}})(\omega_1, \omega_2, 0).
\] (4.2)

The theorem states that the 2D Fourier transform of a projection of an object \( \varphi \) equals to one central slice of the 3D Fourier transform of the object \( \varphi \), where the projection is taken in a direction perpendicular to the slice (Figure 1.2).

It is important to realize that in practice the molecule’s electric potential function \( \varphi \) is of limited spatial extent. On the other hand, numerically it is only possible to compute a finite discrete Fourier transform of \( \varphi \). It is well known that a function with compact support cannot have compactly supported Fourier transform unless it is identically zero. However, this constraint is easily overcome for any finite accuracy [63]. In this chapter, the potential functions \( \varphi \) are assumed to be essentially band-limited to a ball and essentially space-limited to a cube. A ball in the Fourier domain is a natural choice due to the radial symmetry of the CTFs and isotropic treatment of orientations of cryo-EM images. We sample the continuous function \( \varphi \) on a Cartesian grid \( \{ \mathbf{n} : \mathbf{n} \in \mathbb{Z}^3, -N/2 \leq \mathbf{n} < N/2 \} \) to obtain a volume \( V(\mathbf{n}) = \varphi(\mathbf{n}a) \), where
\( a \in \mathbb{R}_+ \) is the grid spacing, and \( N \in \mathbb{Z}_+ \) is large enough to cover the support of \( \varphi \). According to the sampling theorem, we further assume the Nyquist frequency \( 1/(2a) \) is no smaller than half the essential bandwidth of the function \( \varphi \). With the above assumptions, the Fourier projection-slice theorem has the following discretized version: Given a volume \( V \) of size \( N \times N \times N \) with the above assumptions, and a projection’s orientation \( R \in \text{SO}(3) \), define the frequency on the Cartesian grid of a central slice as

\[
\omega = (\omega_1, \omega_2) = \frac{2\pi(k_1, k_2)}{N}, \text{ where } k_1, k_2 \in \mathbb{Z}.
\]

The Fourier projection-slice theorem \( \text{(4.2)} \) implies that the Fourier coefficient at \( \omega \) on the slice is approximated by

\[
\sum_{-N/2 \leq n < N/2} V_n \exp \left( -i \cdot \langle n, R^{-1}(\omega_1, \omega_2, 0) \rangle \right).
\]

(4.3)

In particular if \( \|\omega\| > \pi \), then the Fourier coefficient at \( \omega \) is approximately zero.

Importantly, cryo-EM images are not true projections of a macromolecule because of the effects of the CTFs \[25\]. Mathematically, a CTF is defined as a function in the Fourier domain, which can be approximated by a sinusoidal function depending on the magnitude of the frequency. A CTF has the following form:

\[
\text{CTF} \left( r \right) = \sin \left( -\pi \cdot \left( \text{defocus} \cdot r^2 - \text{Cs} \cdot \lambda^3 \cdot r^4 / 2 \right) - \text{A} \right) \cdot \exp \left( - \left( \frac{r}{2 \cdot \text{B factor}} \right)^2 \right),
\]

where \( r \) is the magnitude of the frequency, \( \text{Cs} \) is the spherical aberration constant in mm, \( \lambda \) is the electron wavelength in picometers, and \( \text{A} \) is amplitude contrast. A cryo-EM image \( I \) is the result of convolving the true projection \( J \) with a point spread function, where the point spread function is the inverse Fourier transform of the CTF \( h \). Thus, following the convolution theorem, \( F(I) = F(J)h \), where \( F \) is the Fourier transform operation.
Denote a CTF as a function \( h : \mathbb{R}_+ \rightarrow \mathbb{R} \), then according to (3), a Fourier slice affected by a CTF is approximated by

\[
\sum_{-N/2 \leq n < N/2} V_n \exp \left( -i \cdot \langle n, R^{-1} (\omega_1, \omega_2, 0) \rangle \right) h(\| \omega \|).
\]

(4.4)

With the knowledge of the Fourier projection-slice theorem and the CTFs, it is natural to define a forward-projector which projects a volume \( V \) to obtain Fourier slices modulated by CTFs.

Consider a volume \( V \) of size \( N \times N \times N \), \( M \) images with corresponding CTFs \((h_1, h_2, \cdots, h_M)\), and rotations \( R_1, R_2, \ldots, R_M \in SO(3) \). For each central slice (2D Fourier transform of images) consider the Cartesian coordinates \( \omega_{k_1,k_2} = (\omega_{k_1}, \omega_{k_2}) = 2\pi(k_1, k_2)/N \), where \( k_1, k_2 \in \mathbb{Z} \). We define a forward-projector \( A \) which projects a volume \( V \) to obtain \( M \) truncated Fourier slices corresponding to the images as

\[
(A(V)) (k_1, k_2, m) = \sum_n V_n \exp \left( -i \cdot \langle n, R_m^{-1} (\omega_{k_1}, \omega_{k_2}, 0) \rangle \right) \cdot h_m (\| \omega_{k_1,k_2} \|),
\]

(4.5)

where \( m = 1, \ldots, M \) is the index of an image, and \( k = (k_1, k_2) \) satisfies the condition

\[
\| k \| \leq N/2 \text{ (inside a ball in the Fourier domain).}
\]

(4.6)

The condition (4.6) is based on the assumption that the function \( \varphi \) corresponding to the volume \( V \) is essentially band-limited to a ball in the Fourier domain. With this definition, the imaging process is modeled as

\[
b = A(V) + \text{noise},
\]

(4.7)

where \( b \) is formed by the 2D discrete Fourier transform of the noisy images and restricting only to frequencies that satisfy (4.6).
4.2.2 The Back-projector $A^*$ and the Toeplitz Structure of $A^*A$

The back-projector $A^*$ is the adjoint operator of $A$. Note that $A^*$ is not equivalent to the inverse of $A$ because of the non-uniform spacing of frequencies. Let $g$ be an arbitrary collection of $M$ truncated slices, that is, $g = (g_{k_1,k_2,m})$, with $1 \leq m \leq M$, and $k = (k_1, k_2)$ satisfy (4.6). From the definition of $A^*$: $\langle A(V), g \rangle = \langle V, A^*(g) \rangle$, we obtain

\[(A^*g)(n) = \sum_{m=1}^{M} \sum_{k} \exp \left(i \cdot \langle n, R_m^{-1}(\omega_{k_1}, \omega_{k_2}, 0) \rangle \right) \cdot h_m (\|\omega_{k_1,k_2}\|) \cdot g_{k_1,k_2,m}. \quad (4.8)\]

The operator $A^*A$ is then given by

\[
A^*A(V)(n) = \sum_{l=1}^{M} V_l \sum_{m=1}^{M} \sum_{k} \exp \left(i \cdot \langle n - l, R_m^{-1}(\omega_{k_1}, \omega_{k_2}, 0) \rangle \right) \cdot h_m (\|\omega_{k_1,k_2}\|)^2 \quad (4.9)
\]

\[
= (Ker * V)(n), \quad (4.10)
\]

where the "convolution kernel" $Ker$ is defined as

\[
Ker(n) = \sum_{m=1}^{M} \sum_{k} \exp \left(i \cdot \langle n, R_m^{-1}(\omega_{k_1}, \omega_{k_2}, 0) \rangle \right) \cdot h_m (\|\omega_{k_1,k_2}\|)^2, \quad (4.11)
\]

and $-N < n = (n_1, n_2, n_3) < N$.

From (4.9) we observe the Toeplitz structure of $A^*A$, which is a Toeplitz matrix of multi-order 3 and of size $N$-by-$N$ (see section 4.1.3). The $(n,l)$ entry of $A^*A$ only depends on $n - l$, that is, $(A^*A)(n,l) = (A^*A)(n - l)$. In fact, from (4.9), we have

\[
(A^*A)(n,l) = \sum_{m=1}^{M} \sum_{k} \exp \left(i \cdot \langle (n - l), R_m^{-1}(\omega_{k_1}, \omega_{k_2}, 0) \rangle \right) \cdot h_m (\|\omega_{k_1,k_2}\|)^2. \quad (4.12)
\]
The Toeplitz structure allows us to rewrite (4.9) as (4.10), that is, as a convolution of the volume $V$ with the kernel $\text{Ker}$, or simply summarized as

$$A^*A(V) = \text{Ker} * V. \quad (4.13)$$

The circular convolution theorem tells us that the Fourier transform of a convolution equals the product of the Fourier transforms. Consider an $n$-by-$n$ Toeplitz matrix $T_n$ and an arbitrary $n$-vector $v$. The matrix-vector multiplication $T_nv$ can be computed by 1D FFTs by first embedding $T_n$ into a $2n$-by-$2n$ circulant matrix $\text{circ}(T_n)$, i.e.,

$$\begin{bmatrix} T_n & U_n \\ L_n & T_n \end{bmatrix} \begin{bmatrix} v \\ 0 \end{bmatrix} = \begin{bmatrix} T_nv \\ L_nv \end{bmatrix},$$

where $U_n$ and $L_n$ are $n$-by-$n$ matrices designed in a way that ensures that the $2n$-by-$2n$ matrix is circulant. Then, the multiplication is carried out by FFTs using the decomposition (4.1) while ignoring the bottom half of the output vector (i.e., $L_nv$). The matrix-vector multiplication thus requires $O(2n \log(2n))$ operations. Similarly, due to the Toeplitz structure of $A^*A$, the matrix-vector multiplication $A^*A(V)$ is a three-dimensional convolution that can be computed using 3D FFTs by embedding the $N$-by-$N$ matrix $A^*A$ into a $2N$-by-$2N$ circulant matrix $\text{circ}(A)$ of multi-order 3, and then carrying out the computation by using the 3D version of (4.1) for decomposing $\text{circ}(A)$ (see section 4.1.3). Using the property (4.13), it can be verified that the first “column” of $\text{circ}(A)$ is

$$C_{1,1}^A = \text{Ker}(c(i_1), c(i_2), c(i_3)), \quad (4.14)$$
where \(1 \leq i = (i_1, i_2, i_3) \leq 2N\) and the function \(c\) is defined as
\[
c(i) = \begin{cases} 
  i, & 1 \leq i \leq N \\
  1, & i = N + 1 \\
  i - 2N, & N + 2 \leq i \leq 2N 
\end{cases}.
\] (4.15)

The computation of \(A^* A (V)\) thus requires \(O(8N^3 \log (8N^3))\) operations.

### 4.2.3 The Conjugate Gradient (CG) Method

We reconstruct the volume by minimizing the cost function
\[
\rho (V) = \| b - A (V) \|^2, \tag{4.16}
\]
where \(b\) includes all measured Fourier slices, \(A\) is the forward-projector, given by (4.5), and \(V\) is the unknown volume. A solution to the minimization problem can be found by setting to zero the derivative of \(\rho\) with respect to \(V\), yielding
\[
\]

Since \(A^* A\) is symmetric positive-semidefinite, we can apply CG to find the minimizer to the cost function (4.16).

To be used in an iterative method, the operation of \(A^* A\) must be extremely efficient. However, applying \(A\) and \(A^*\) separately at each iteration is time-consuming since the time cost of one application of either \(A\) or \(A^*\) is equivalent to the cost of one application of NUFFT, whose time complexity is \(O(MN^2 \max (\log M, \log (N^2)))\) \([21, 32, 24]\). However, this efficiency problem can be overcome using the property (4.13). Thus, applying \(A^* A\) (or equivalently \(\text{Ker}\)) to a vector requires \(O(8N^3 \log (8N^3))\) operations. Note that both \(\text{Ker}\) and \(A^* b\) can be precomputed using NUFFT only.
once before applying the CG method. More details about the time complexity of the computation of \( \text{Ker} \) and \( A^*b \) are provided in section 4.2.4.

The convergence rate of the CG method has been well studied (see [61]), and it depends on how clustered the spectrum of \( A^*A \) is. The output of the projector \( A \) has no high frequency information outside a ball in the Fourier domain, resulting in an extremely large condition number of \( A^*A \) and very small eigenvalues in the spectrum. The ill-conditioning of the reconstruction problem causes the semi-convergence behavior [73, 36, 35], which can be characterized as initial convergence toward the exact solution and later divergence. From the perspective of regularization, the updated estimated volume in each iteration is a regularized solution and the number of iterations plays the role of the regularization parameter. The iteration count controls the compromise between the signal-to-noise ratio (SNR) and the residual aliasing artifact. The basic principle of regularization is to smooth the solution by truncating or damping the small eigenvalue components. The initial iterations pick up the eigenvalue components corresponding to the largest eigenvalues. As the iteration number increases, more and more small eigenvalues are captured and the degree of regularization decreases. As a result, the residual norm declines sharply at early stages of the iterative process before it levels off. For a well behaved reconstruction, the plot of the residual norm in \( \log_{10} \) scale versus the iteration count generally exhibits an L-curve characteristic, as shown in Figure 4.1. In this manner, the iteration procedure can be divided into 3 phases. On the left of the L-curve, the residual norm declines very fast and we refer to it as a dropping phase; on the right side, the residual norm levels off and it is a level phase. The L-corner then represents the transition phase, where the noise and artifacts are usually well compromised. However, it is practically difficult to locate the point where SNR and artifacts are optimally compromised and best reconstructed volume quality is achieved. It is recommended to pick a num-
4.2.4 Fourier-based Iterative Reconstruction Method (FIRM)

The input to our reconstruction algorithm is the following:

1. $M$ projection images of size $N \times N$ denoted $I_1, I_2, \ldots, I_M$ of an unknown volume $V$ of size $N \times N \times N$.

2. The orientations of the images $R_1, R_2, \ldots, R_M \in \text{SO}(3)$.

3. The CTFs $h_1, h_2, \ldots, h_M$.

The description of FIRM is as follows:

1. Compute the 2D DFT of all images using 2D FFT. Truncate the square Fourier slices and form the vector $b$ in (4.7). ($O(MN^2 \log (N^2))$)

2. Compute the back-projection $A^*b$ according to (4.8) using NUFFT. ($O(MN^2 \max (\log M, \log (N^2)))$)
3. Compute the convolution kernel $\mathbf{Ker}$ according to (4.11) using NUFFT.

\[ (O(4MN^2 \max (\log M, \log (N^2)))) \]

4. Use CG with input $\mathbf{A}^*\mathbf{b}$, $\mathbf{Ker}$ and an initial guess (all-zero volume). The output is the estimated volume. ($O(8N^3 \log (8N^3))$ operations for each iteration)

The running time of the algorithm is dominated by Steps 2-3. The time complexity of Step 2 for the NUFFT algorithm is obtained from [21, 32, 24]. In Step 3, although $\mathbf{Ker}$ is about 8 times as large as the original volume (i.e., a factor of 2 in each dimension), by noting that $\mathbf{Ker}(-\mathbf{n}) = \mathbf{Ker}(\mathbf{n})$, the time cost of computing $\mathbf{Ker}$ is about 4 times the cost of computing the back-projection $\mathbf{A}^*\mathbf{b}$. Using property (4.13), the computation in each iteration of CG in Step 4 is efficiently reduced to FFTs and matrix-matrix point-wise multiplication. Moreover, it is easy to parallelize the computation of $\mathbf{A}^*\mathbf{b}$ and $\mathbf{Ker}$ by noting that both of them are summation over back-projection of a single projection. A MATLAB package for FIRM is available to download through the website http://www.math.princeton.edu/~lanhuiw/software.html.

### 4.3 Numerical Results

We implemented FIRM using the MATLAB programming Language. The NUFFT package provided by [24] is used for precomputation of back-projections and convolution kernels. We compare FIRM with 4NN implemented within the framework of the SPARX image processing system [41]. The numerical experiment is performed on a machine with 2 Intel(R) Xeon(R) CPUs X5570, each with 4 cores, running at 2.93 GHz. Both MATLAB and SPARX are limited to single core computations.

In the numerical experiment, a 50S ribosomal subunit volume of size $90 \times 90 \times 90$ (Figure 1.2 Left) is used to generate projections. We use SPARX to generate a random conical tilt series consisting of 10,000 simulated projections. The tilt angle is fixed to be 60° and the azimuthal angles are randomly sampled from the uniform distribution
over $[0^\circ, 360^\circ]$. Thus there is a missing cone in the coverage of Fourier space by the slices. The purpose to use a random conical tilt series is that not only the accuracy of the reconstructed volumes excluding the missing cone can be studied, but also the artifacts of the volumes inside the missing cone can be observed. For a real dataset in cryo-EM, the geometry of the collected images cannot be totally controlled. Thus the artifacts of the reconstructions due to the uneven sampling are of interest.

The 10,000 projections are divided randomly to 3 defocus groups and filtered by the CTFs which are generated with parameters detailed in the caption of Figure 1.3. We refer the CTF filtered projections as clean images. The noisy images are generated by adding white Gaussian noise to the clean images. In this experiment, the SNR of the noisy image is set to 1.

To evaluate the accuracy of the reconstructions, we use the 3D Fourier Shell Correlation (FSC) in (3.72). For each reconstructed volume, we measure its FSC against the clean 50S ribosomal subunit volume, that is, in our measurement $V_1$ is the reconstructed volume, and $V_2$ is the “ground truth” volume. In this case, FSC is also called Fourier Cross-Resolution (FCR). To measure the accuracy of the reconstructed volumes outside and inside the missing cone respectively, we use modified FCR for the target Fourier volumes excluding or within the missing cone region.

FIRM is compared with other reconstruction algorithms (GDFR, SIRT, 4NN, etc). However we only report the comparison with 4NN since it performed best in terms of accuracy and running time. The reconstructed volumes by FIRM are the estimations in 30th iteration for both the clean and noisy image datasets, where the “transition phase” is in the L-curve (see details in section 4.2.3). FIRM spent 4 seconds on FFTs of the images, 293 seconds on back-projection, 1143 seconds on computing the kernel, and 1 second on each CG iteration. The total time cost by FIRM is around 1470 seconds, which is about five times the time cost by 4NN (290 seconds).
Figure 4.2: FCRs/FSCs of the reconstructed volumes excluding or within the missing cone region. (a) and (b) are comparison between reconstructions from clean images. (c) and (d) are comparison between reconstructions from noisy images with SNR=1.

seconds). From Figure 4.2 we observe that the accuracy of the reconstructions by the two algorithms are almost the same excluding the missing cone. However, the measurement inside the missing cone demonstrates that there is less artifacts created by FIRM than by 4NN.

4.4 Summary and Discussion

In this chapter, we propose a fast and accurate Fourier-based iterative reconstruction method (FIRM) to reconstruct molecular structures from cryo-EM images. To study the imaging process in cryo-EM, we define a forward-projector $A$ which converts a given volume to Fourier slices affected by CTFs. Therefore, the imaging model is
\[ b = A(V) + \text{noise} \] where \( A \) is the forward-projector, \( V \) is the unknown volume we are interested in, and \( b \) is the measurement of the Fourier slices. To reconstruct the volume \( V \) from the measurement \( b \), CG is applied to estimate the reconstructed volume by minimizing \( \rho(V) = \|b - A(V)\|^2 \). The solution is found by setting the derivative of \( \rho \) to zero, yielding the equation \( A^*A(V) = A^*b \). \( A^*A \) has Toeplitz structure and thus \( A^*A(V) = \text{Ker} \ast V \), where \( \text{Ker} \) is a convolution kernel. Using this property, which is key to our method, the computation of \( A^*A(V) \) is reduced to FFTs and matrix-matrix point-wise multiplication according to the convolution theorem. As a result, the computation of each CG iteration is fast.

The main advantage of iterative methods (ART, SIRT and FIRM) in general is their applicability to diverse data collection geometries and to data with uneven distribution of projection directions. However, for ART and SIRT, the computational requirements are dominated by the back-projection steps and thus their running time exceeds that of other algorithms (WBP1, WBP2, GDFR and 4NN) for the typical number of required iterations (typically 10-200). Instead of back-projecting in each iteration, FIRM computes the back-projection only once in the pre-computation stage that also includes the computation of the kernel using NUFFT. The most time cost by FIRM is thus at the stage of the preparation before CG. The time cost of iterations in FIRM is negligible compared to the pre-computation.

The numerical experiments demonstrate that compared with 4NN, FIRM is fast and accurate, and it performs well in merging information from different defocus groups. Moreover, FIRM creates a satisfactory reconstruction in the case of a missing cone in Fourier space with less artifacts than 4NN. As of the running time, FIRM costs about five times the time used by 4NN, which is the fast direct inversion algorithm in SPARX. We remark that FIRM is flexible in the sense that it can incorporate other regularization terms that can be naturally formulated in Fourier space (e.g., damping high frequencies), or other prior knowledge about the volume, such as positivity.
constraints. In severe situations when images correspond to only a small number of views or when there are some gaps in Fourier space, regularization can alleviate the ill-conditioning of the problem. In [105] we demonstrated how the computational framework of FIRM can be modified to treat a regularization term that consists of the $\ell^1$ norm of the wavelet expansion coefficients of the volume. The purpose of such regularization terms is to promote sparsity in the expansion of the volume in the wavelet basis. We remark that other regularization terms involving the total variation functional [48] or tight frame expansions are also possible.
Conclusion and Future Work

Our contribution [108] to the synchronization problem is three-fold: First, we introduce a robust penalty function involving the sum of unsquared deviations and derive a relaxation that leads to a convex optimization problem; Second, we apply the ADMM to minimize the penalty function; Finally, under a specific model of the measurement noise and for both complete and random measurement graphs, we prove that the rotations are exactly and stably recovered, exhibiting a phase transition behavior in terms of the proportion of noisy measurements. Numerical simulations confirm the phase transition behavior for our method as well as its improved accuracy compared to existing methods.

We plan to continue to generalize the LUD framework to more general measurement graphs, and to other problems such as the orthogonal Procrustes problem [95]. More investigation is needed for the exact and stable recovery conditions for LUD for general graphs, where it is speculated that the second eigenvalue of the graph Laplacian plays a role in determining the location of a phase transition of rotation recovery [6]. Another direction of the future work is to extend LUD in a straightforward manner for the orthogonal Procrustes problem of rotating $n$ matrices toward a best least-unsquared deviations fit.
In chapter 3 and chapter 4, we proposed robust and fast algorithms (LUD and FIRM) to improve the common-lines based reconstruction methods in the two stages: orientation determination, and inversion from the cryo-EM images.

Motivated by the successful application of LUD in the synchronization problem, we applied LUD method in a similar manner to solve the problem of orientation determination from cryo-EM images. In order to prevent artificial clustering of the estimated viewing directions, we further introduced a spectral norm term that is added as a constraint or as a regularization term to the relaxed minimization problem. The resulted problems are solved by using either the alternating direction method of multipliers (ADMM) or an iteratively reweighted least squares procedure (IRLS). Numerical experiments with both simulated and real images demonstrate that the proposed methods significantly reduce the orientation estimation error when the detection rate of common-lines is low. It is also possible to consider other self consistency errors involving the unsquared deviations raised to some power $p < 1$ (the cases $p = 1, 2$ correspond to LUD and LS, respectively).

To estimate the 3D model from the 2D projection images, we proposed a fast and accurate Fourier-based Iterative Reconstruction Method (FIRM). We take advantage of the Toeplitz structure of the iteration operator, which allows for a computationally effective implementation of the algorithm. Furthermore, we precompute recurring quantities using non-uniform FFTs. The iterations by FIRM are considerably faster than those of traditional iterative algebraic approaches, while maintaining the same accuracy. The time complexity of FIRM is comparable to the direct Fourier inversion method. Moreover, FIRM is flexible to incorporate CTF correction and regularizations. The application of FIRM can be extended to the reconstruction problem in cryo-electron tomography, where the volume size of the 3D structure is about ten times larger than that in cryo-EM, while the number of images are about 100 or 1000 times smaller than that in cryo-EM.
Another challenge in orientation estimation is detection of common-lines between images. Currently, we apply low-pass filter or principal component analysis (PCA) based filter to denoise the radial lines of Fourier transforms of cryo-EM images, and then detect the common-lines by finding maximum normalized cross-correlation between radial lines for each pair of images. It is possible to improve the common-line detection rate by using PCA and Wiener filter methodology, and we can show that the combination of PCA and Wiener filter is equivalent to the maximum likelihood (ML) method. Moreover, this new common-line detection approach based on ML can be applied to images in different defocus groups.

We proposed LUD based algorithms to determine orientations of cryo-EM images from common-lines between them. When the common-line detection rate was low due to the high noise level of the projection images, we discovered that the LUD based algorithms outperformed the LS based algorithms in orientation estimation and resulted in better reconstructed ab-initio models. However, it is possible that other methods that use more information from the projection images would perform better. Besides the information of common-lines, we can make use of information of the “non-common lines” between the projection images. The “non-common lines” are the radial lines in the polar Fourier transforms of a pair of projection images where they do not intersect with each other. Each pair of cryo-EM images with different projection directions has only one pair of common-lines, while it has many more pairs of “non-common lines”. The “non-common lines” contain rotational information of images that can be used as penalties or constraints in the problem of orientation estimation, i.e., we do not need to consider certain configurations of the orientations that mistake “non-common lines” for common-lines. ML method can be used to find a configuration of orientations that minimizes the sum of squared distances between pairs of radial lines, where the pairs of radial lines are determined as common-lines by the configuration of orientations. In this manner, the configurations of orientations
that determine dissimilar radial lines as common-lines are penalized, where the dis-
similar radial lines are very likely to be “non-common lines”. However, the difficulty
lies in the size of the minimization problem due to the added “non-common lines”
information. More study is needed if one wants to exploit the “non-common lines”
information for a more accurate and robust orientation estimation algorithm.
Bibliography


[53] R. Marabini, G. T. Herman, and J. M. Carazo. 3D reconstruction in electron
microscopy using ART with smooth spherically symmetric volume elements

[54] D. Martinec and T. Padjla. Robust rotation and translation estimation in
multiview reconstruction. In IEEE Conference on Computer Vision and Pattern

[55] F. Mezzadri. How to generate random matrices from the classical compact


[57] K. Mohan and M. Fazel. Iterative reweighted algorithms for matrix rank mini-

[58] A. Naor, O. Regev, and T. Vidick. Efficient rounding for the noncommutative


[60] A. Nemirovski. Sums of random symmetric matrices and quadratic optimiza-
2007.

USA, 2000.


[64] P. Penczek. Chapter One - Fundamentals of Three-Dimensional Reconstruction
from Projections. In Cryo-EM, Part B: 3-D Reconstruction, volume 482 of

[65] P. Penczek, R. Grassucci, and J. Frank. The ribosome at improved resolution:
New techniques for merging and orientation refinement in 3D cryo-electron mi-

[66] P. Penczek, M. Radermacher, and J. Frank. Three-dimensional reconstruction


