Protein structural calculation from NMR spectroscopy

Yuehaw Khoo

A Dissertation

Presented to the Faculty

of Princeton University

in Candidacy for the Degree

of Doctor of Philosophy

Recommended for Acceptance

by the Department of

Physics

Adviser: Amit Singer and Michael Aizenman

September 2016

Abstract

NMR spectroscopy has been used to determine more than 11000 protein structures in the Protein Data Bank (PDB). By providing geometric constraints between pairs of nuclei in the protein, from NMR spectroscopy the 3D structure of the protein can be obtained. The best established structural calculation methods use distance restraints between pairs of hydrogen atoms provided by nuclear Overhauser effect (NOE). Since the extraction of distance restraints from NOE can be challenging for large protein, the use of the residual dipolar coupling (RDC) has become a popular alternative for protein structuring.

In this thesis, integrated structural calculation approaches using restraints provided by both NOE and RDC are studied. Since the optimization problems arise in structural calculation are non-convex in nature, we devise convex relaxation methods to obtain the global optimum of these problems. This is in contrast to traditional optimization approaches such as simulated annealing that lack the guarantees of global optimality. In the first part of the thesis, we present a divide-and-conquer approach to solve the structural determination problem from distance restraints. In this approach, small fragments of the molecules are first built from distance restraints. Then in Chapter 2, a global registration method is developed to stitch the small fragments into a global structure. Such divide-and-conquer approach can shorten the running-time via parallel computing. However, the divide-and-conquer approach is essentially a distance based procedure in that it only uses distance restraints to build each fragment. Therefore in Chapter 3, we present an integrated approach that directly uses both RDC and NOE to construct the 3D structure of the protein with high accuracy. We apply our method to the protein ubiquitin and obtain structure with 1 Å resolution. In Chapter 4, we describe a method for Saupe tensor estimation from RDC when having multiple protein fragments. In particular, we study the bias arises in Saupe tensor estimation from the structural noise of the protein fragments. We show how Saupe tensor estimation can

be used to enhance the global registration method by aligning the small fragments to a principal axis frame.

Acknowledgements

First and foremost I would like to express my deepest gratitude to my adviser Amit Singer. His scientific vision and broad mathematical knowledge that spans geometry, analysis, optimization and statistics have been a great inspiration to me. I have only managed to apply a small subset of his knowledge in this thesis. By the time I am close to graduation, I look back and wish I would have learnt more from him. He also provides numerous helps career-wise by connecting me with established researchers and informing me about interesting workshops and seminars. He even helps and advises me in making slides and posters, writing research statements, making personal website and choosing career path. I feel extremely fortunate to have an adviser like Amit who supports his students in such a multitude of ways.

I am grateful to Michael Aizenman for his support of my research outside of physics department, ensuring that I stayed on track during my doctoral study and reading my thesis. I also want to thank David Cowburn at Albert Einstein College of Medicine for sharing with me the challenges and new advancements in NMR spectroscopy. He teaches me how to identify relevant problems in NMR spectroscopy, and many of his insights inspire this thesis and other ongoing works. I specially want to thank Phuan Ong for advising me in experimental condensed matter physics during my first two years in Princeton and even supporting my decision to switch out of his group. I will miss the days he exposed deep physics theory in a simple way that I can understand on the hallway blackboards. I am also honored to have Joshua Shaevitz serving as my thesis committee member.

I want to take this opportunity to thank my collaborators, Kunal Chaudhury (now at Indian Institute of Science) and Jose Simoes Bravo Ferreira. I cannot imagine working on the NMR problem without their helps and company during my time at Princeton. I want to thank every wonderful person in Amit's group, especially Joakim Anden, Afonso Bandeira, Nicolas Boumal, Yutong Chen, William Leeb, Onur Ozyesil, and Justin

Solomon, for all the stimulating discussions that shapes the way I think about a problem. I am also grateful to Tian Liang who spent many hours patiently helping me to understand various physics concepts when I first started my graduate study. I also want to thank my colleagues during my internship at Siemens, Ankur Kapoor and Ahmet Tuysuzoglu, who introduced me to interesting problems in the field of medical imaging.

Special thanks go to members in Princeton Christian Church for helping me to grow spiritually. I really appreciate Shaoliang Zhang and Qinqin Shi for providing me a place to stay and taking care of me like a member of their family over the last year. I also want to thank my parents, my siblings and my grandmother for their support and their understanding when I go home too little during the course of my doctoral study. I am always grateful for having a wife, Lucy, who accompanies me in every joyful or difficult moment. Indeed, without her encouragement I would not have even entered graduate school and persevered through my study. Finally, I am most thankful for having God who knows and sympathizes my weaknesses, in whom I can take refuge.

To my family.

Contents

List of plots

Tables

Notations

We first summarize the notations that will be used throughout the thesis. We use upper case letters such as A to denote matrices, and lower case letters such as a for vectors. We use I_d to denote the identity matrix of size $d \times d$. We denote the diagonal matrix of size $n \times n$ with diagonal elements $c_1, \ldots, c_n$ as $\mathrm{diag}(c_1, \ldots, c_n)$. We will frequently use block matrices built from smaller matrices. For some block matrix A, we will use A_{ij} to denote its (i, j)-th block. The size of each block will be made clear in the context. For a matrix A, we use $A(p, q)$ to denote its (p, q)-th element. We use $A \succeq 0$ to mean that A is positive semidefinite, that is, $u^T A u \geq 0$ for all u. We use $\mathbb{O}(d)$ and $\mathbb{SO}(d)$ to denote the group of orthogonal matrices and special orthogonal matrices acting on $\mathbb{R}^d$, We use $\|x\|$ to denote Euclidean norm of $x \in \mathbb{R}^n$ (n will usually be clear from the context, and will be pointed out if this is not so). We denote the trace of a square matrix A by $\mathrm{Tr}(A)$. The Frobenius and spectral norms are defined as

$$\|A\|_F = \mathrm{Tr}(A^T A)^{1/2} \quad \text{and} \quad \|A\|_{\mathrm{sp}} = \max_{\|x\| \leq 1} \|Ax\|.$$

The Kronecker product between matrices A and B is denoted by $A \otimes B$ [57]. The all-ones vector is denoted by $\mathbf{1}$ (the dimension will be obvious from the context).

Chapter 1

Introduction

NMR spectroscopy has been used to determine more than 11000 protein structures in the Protein Data Bank (PDB) [13]. In the nutshell, NMR spectroscopy experiments provide geometric constraints between pairs of nuclei in the protein. With these pairwise constraints one can hopefully determine the 3D structure of the protein. As demonstrated in Figure 1.1, the typical NMR structural determination pipeline consists of the following steps: (1) Peak picking from NMR spectra, (2) chemical shift assignment (spectral assignment), (3) assignment of geometric restraints, and (4) structural calculation [68]. However, unlike the case of X-ray crystallography, for NMR spectroscopy, the process of going from experimental spectra to final 3D structure is not yet close to being fully automated. This is largely due to the fact that there are typically ambiguities in identifying peaks and assigning the chemical shifts, leading to constraints being placed wrongly on pairs of atoms. Based on structures calculated from geometric constraints, experts judgement is needed to correct peak lists and spectral assignment iteratively. This process is not only tedious, but also allows the subjectivity of the practitioners to influence the final protein structure [160, 68].

This thesis focuses on the step of calculating protein structure given the geometric restraints. Although this step has a long history of being automated [39] and is generally

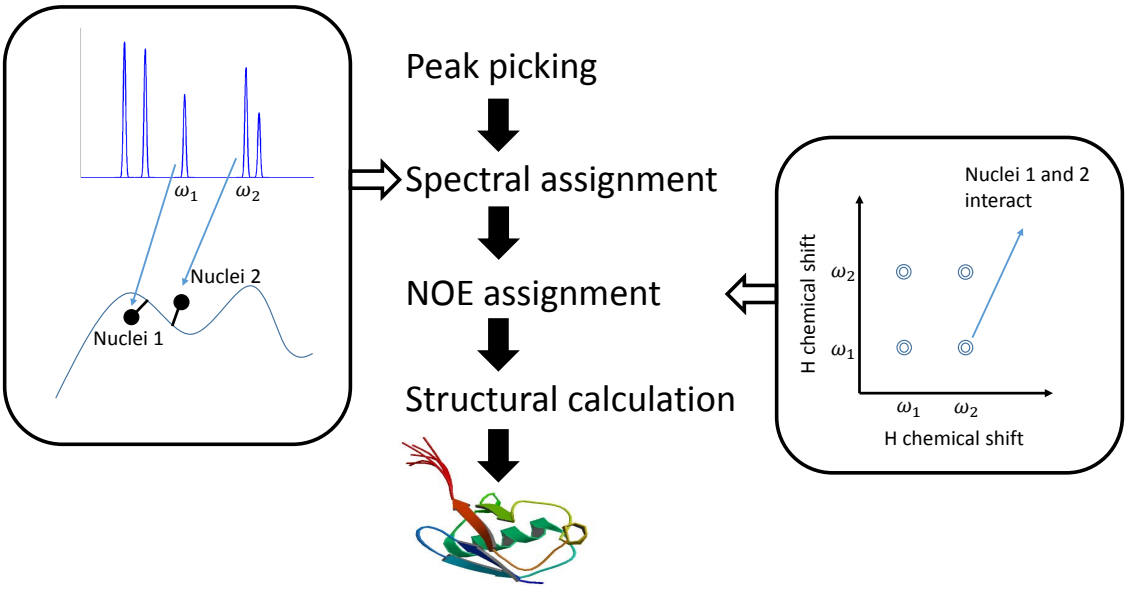


Figure 1.1: The full procedure for protein structural determination. After picking the resonance peaks, the chemical shift assignment procedure labels each nuclei in the protein with experimentally measured resonance frequencies. Then the restraint assignment procedure builds the interaction network of the nuclei from off-diagonal peaks in 2D or 3D NMR spectra after chemical shift assignment. The resulting geometric restraints between pairs of nuclei are then used as input for the structural calculation procedure.

considered as an “easier” problem than its preceding spectral assignment problem, improvement can still be made in terms of accuracy, speed, and its ability to include various types of data. Such improvements will in turn benefit the entire structural determination pipeline. Since it is often necessary to iterate between correcting spectral assignment and re-calculating the protein structure, being able to calculate a structure rapidly enables quick validation of the assignments. More recent measurements such as residual dipolar coupling are also calling for protein structuring algorithms that can handle them more efficiently than traditional methods based on simulated annealing. The usage of RDC can greatly alleviate the computational burden in obtaining unambiguous distance restraints assignment and is therefore of immediate interest to NMR practitioners. From a mathematical point of view, the global optimization approaches that are widely used in protein structural calculation generally lack the guarantees of obtaining global optimality in polynomial time. Since the functions we are optimizing are typically non-convex, these methods can fail to attain the “ground truth” protein structure even when supplied with synthetic noiseless data. In this thesis, we aim to bring improvements in accuracy and speed to the structural calculation procedure through the use of convex programming relaxations. The main idea of such technique is to replace the non-convex optimization problem with a convex surrogate problem. When the global optimums of the convex problem and original non-convex problem coincide, the solution to the non-convex problem can be retrieved efficiently via convex optimization.

1.1 The structural calculation problem

Before NMR spectroscopy experiments, we assume the protein sequence and its covalent structure are known. This means the amino-acid type for each residual and the bond length, bond angles are given. In NMR spectroscopy, different experiments can be devised to measured different interactions between pairs of nuclei. Among all interactions the

most commonly measured ones are the Nuclear Overhauser Effect (NOE) that provides hydrogen-hydrogen distances, J-coupling that provides torsion angles and residual dipolar coupling (RDC) that provides bond orientations. In this section we describe the NOE and RDC in detail.

The NOE and RDC measurements are based on the dipole-dipole interactions between pairs of nuclei. For two spin operators $\mathbf{I}_1, \mathbf{I}_2$, the energy of dipolar interaction can be described by the Hamiltonian

$$H_D = \frac{\mu_0 \gamma_m \gamma_n \hbar^2}{4\pi d_{mn}^3} (3 \frac{(\mathbf{I}_m \cdot \mathbf{d}_{mn})(\mathbf{I}_n \cdot \mathbf{d}_{mn})}{d_{nm}^2} - \mathbf{I}_m \mathbf{I}_n) \tag{1.1}$$

where $\mathbf{d}_{mn}$ denotes the displacement vector between nuclei m and n, d_{mn} denotes the magnitude of displacement, γ_m, γ_n denotes the gyromagnetic ratios of the two nuclei. This complicated looking Hamiltonian admits simple expression in the limit of high magnetic field B, which is the case in a typical NMR experiment. In the high-field limit, the Hamiltonian H_D can be treated as a perturbation to the Zeeman splitting

$$H_Z \propto I_{mz} B + I_{nz} B. \tag{1.2}$$

using perturbation theory [95]. In this case, the components in H_D that do not commute with H_Z can be dropped to obtain the first order perturbation to H_Z [48, 78]. Assuming the global magnetic field points in the z-direction, the dipolar coupling can be approximated as

$$H_D \approx \frac{\mu_0 \gamma_m \gamma_n \hbar^2}{4\pi d_{mn}^3} (3\cos(\theta_{mn})^2 - 1)(3 I_{mz} I_{nz} - \frac{1}{4}(I_m^+ I_n^- + I_m^- I_n^+)) \tag{1.3}$$

where I_j^+, I_j^- are the spin raising and lowering operators respectively and θ_{mn} is the angle between $\mathbf{d}_{mn}$ and the global magnetic field.

1.1.1 Nuclear Overhauser effect (NOE)

As molecules tumble in the solution, the spatial component $(3\cos(\theta_{mn})^2-1)$ of the dipolar interaction H_D averages to zero. However, the fluctuation of the dipolar interaction induces *cross-relaxation* between the magnetization of nuclei m and n. Such transition can be observed in the nuclear Overhauser effect spectroscopy (NOESY). The main idea of NOESY is to invert the magnetization of the nuclei at equilibrium. Then the non-equilibrium magnetization relaxes back to the original equilibrium state through transition induced by time-dependent dipolar coupling. Details of such experiment can be found in the excellent textbook [86, Chapter 8]. The exchange of magnetization between two nuclei in close proximity can be observed via cross-peaks in the NOE spectra. In Figure 1.2, we give an example of a 2D NOE spectra of hydrogen nuclei. In such type of spectra, an off-diagonal peak at (ω_m, ω_n) indicates the existence of cross relaxation between a pair of hydrogens with chemical shift ω_m and ω_n. However, the NOESY can only provide short range distance measurements, typically for pairs of hydrogens within 5 Å due to the $1/d_{mn}^6$ dependence of the cross-peak intensity. Since the dipolar coupling depends on $1/d_{nm}^3$, the fluctuation of dipolar coupling in time gives rise to cross relaxation rate proportional to $1/d_{mn}^6$ [113], which is a consequence of the time-dependent perturbation theory [118, 93].

From the distances provided by NOESY cross-peaks, the best-established structural determination methods solve variations of the following constraint satisfaction problem

$$\text{Find} \quad x_1, \ldots, x_K \in \mathbb{R}^3 \quad \text{such that} \quad d_{ij}^{\text{low}} \leq \|x_i - x_j\| \leq d_{ij}^{\text{up}}. \tag{1.4}$$

in order to obtain the 3D coordinates of all K atoms in a protein. We note that since torsion angle restraints can be regarded as distance restraints for nuclei that are three bonds apart [30], it can be included in problem (1.4) just like NOE. The solution of (1.4) can only be determined up to a rigid transformation, as any rigid transformation

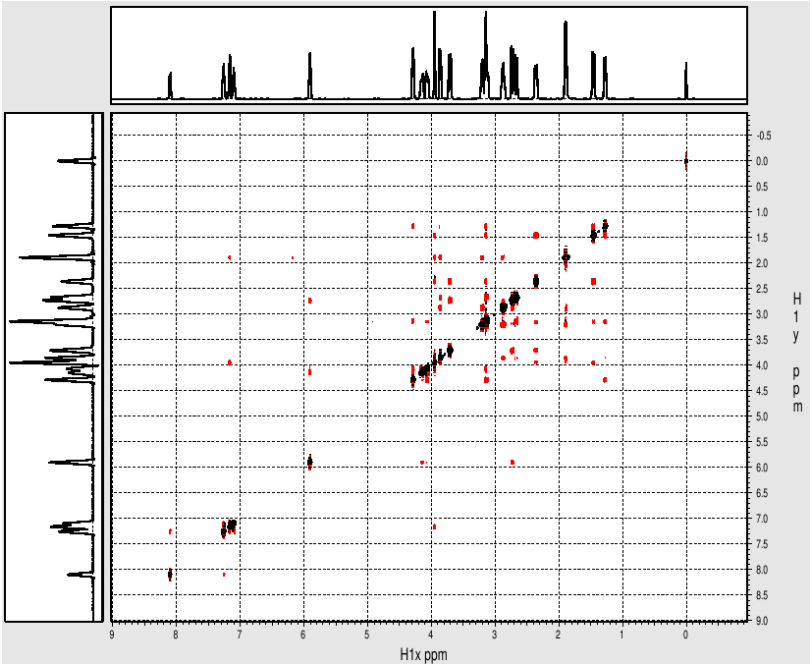


Figure 1.2: Example of a 2D NOE spectra. The image is downloaded from [171].

preserves the pairwise distances.

1.1.2 Residual dipolar coupling (RDC)

The RDC of molecules can be measured when the molecule ensemble in solution exhibits partial alignment with the magnetic field in a NMR experiment. In this case, the term $(3\cos(\theta_{mn})^2 - 1)$ in (1.3) does not average to zero. Due to the $1/d_{mn}^3$ dependence, such effect can be measured with high precision and provides alignment information of a specific bond to the magnetic field. This is in contrast to the NOE with $1/d_{mn}^6$ dependence. Hence the importance of RDC in obtaining high quality protein structures and studying molecular dynamics has increased considerably over the last decade. For a detailed survey of RDC and its applications we refer readers to [100, 143].

Let v_{nm} be the unit vector denoting the direction of the bond between nuclei n and m. Let b be the unit vector denoting the direction of the magnetic field. The RDC RDC_{nm}

due to the interaction between nuclei n and m is

$$\mathrm{RDC}_{mn} = D_0 \left\langle \frac{3(b^T v_{mn})^2 - 1}{2} \right\rangle_{t,e} \tag{1.5}$$

where

$$D_0 = -\frac{\gamma_n \gamma_m h}{2\pi^2 d_{nm}^3}, \tag{1.6}$$

and $\langle \cdot \rangle_{t,e}$ denotes the ensemble and time averaging operator. The quantity RDC_{mn} provides the strength of dipolar coupling (1.3) when molecules are partially aligned. Since the spectrum of the original Zeeman Hamiltonian H_Z splits further under $< H_D >_{t,e}$ by an amount proportional to RDC_{mn}, the RDC can be conveniently measured by the splitting of resonance peaks in NMR spectra. As presented, RDC depends on the relative angle between the magnetic field and the bond. In principle, extracting such angular information from RDC could complement NOE and possibly other measurements for determining the molecular structure.

It is conventional to interpret the RDC measurement in the molecular frame. More precisely, we treat the molecule as being static in some coordinate system, and the magnetic field direction being a time and sample varying vector. In this case the RDC becomes

$$\mathrm{RDC}_{nm} = D_0 v_{nm}^T S v_{nm}, \tag{1.7}$$

where the Saupe tensor S is defined as

$$S = \frac{1}{2}(3B - I_3), \qquad B = \left\langle b b^T \right\rangle_{t,e}. \tag{1.8}$$

B is known as the field tensor and I_3 denotes the 3×3 identity matrix. We note that S is symmetric and $\mathrm{Tr}(S) = 0$. In order to use RDC for structural calculation or refinement of a protein, S is usually first determined from a known structure (known v_{nm}) that is similar to the protein. To satisfy the assumption that the molecule is static in the

molecular frame, a rigid fragment of the known structure has to be selected. S can be determined if the fragment contains sufficiently many RDC measurements. Here we discuss the singular value decomposition (SVD) approach [101] for estimating the Saupe tensor that will be used many times in this thesis. Using the fact that S is symmetric and $\mathrm{Tr}(S) = 0$, eq. (1.7) can be rewritten as

$$\begin{aligned}\mathrm{RDC}_{nm}/D_0 = (\nu_{nm_y}^2 - \nu_{nm_x}^2)S_{yy} + (\nu_{nm_z}^2 - \nu_{nm_x}^2)S_{zz} \\ + 2\nu_{nm_x}\nu_{nm_y}S_{xy} + 2\nu_{nm_x}\nu_{nm_z}S_{xz} + 2\nu_{nm_y}\nu_{nm_z}S_{yz},\end{aligned} \quad (1.9)$$

where ν_{nmi}, $i = x, y, z$ are the different components of ν_{nm} in the molecular frame. Hereafter we let $r_{nm} = \mathrm{RDC}_{nm}/D_0$, to which we refer as the RDC measurements. When there are M RDC measurements, eq. (1.9) results in M linear equations in five unknowns ($S_{yy}, S_{zz}, S_{xy}, S_{xz}$ and S_{yz}), that can be written in matrix form as

$$As = r, \qquad s = \begin{bmatrix} S_{yy} \\ S_{zz} \\ S_{xy} \\ S_{xz} \\ S_{yz} \end{bmatrix} \in \mathbb{R}^5, \qquad r = \begin{bmatrix} r_{n_1 m_1} \\ \vdots \\ r_{n_M m_M} \end{bmatrix} \in \mathbb{R}^M, \quad (1.10)$$

and $A \in \mathbb{R}^{M\times 5}$.

Let the SVD of matrix A be

$$A = U\Sigma V^T, \quad (1.11)$$

where $U \in \mathbb{R}^{M\times 5}$ is a column orthogonal matrix (i.e. $U^T U = I_5$), $V \in \mathbb{R}^{5\times 5}$ is an orthogonal matrix, and $\Sigma \in \mathbb{R}^{5\times 5}$ is a positive diagonal matrix. We assume that $M \geq 5$ and that A has full rank for otherwise there is no unique solution to the linear system

(1.10). The estimator of the Saupe tensor entries s proposed in [101] is

$$\hat{s} = V\Sigma^{-1}U^T r. \tag{1.12}$$

This is equivalent to the ordinary least squares (OLS) solution to the linear system (1.10), given by

$$\hat{s} = (A^T A)^{-1}A^T r. \tag{1.13}$$

For this reason, we frequently refer to the SVD method for Saupe tensor estimation as the OLS method. The computational aspects of employing the expressions in (1.12) and (1.13) are discussed in [161]. Notice that the Saupe tensor estimator given by (1.12) and (1.13), denoted $\hat{S}$, is the solution to the optimization problem

$$\min_{S} \sum_{i=1}^{M} |r_{n_i m_i} - v_{n_i m_i}^T S v_{n_i m_i}|^2 \quad s.t.\ S \text{ is symmetric, } \mathrm{Tr}(S) = 0. \tag{1.14}$$

As such, the OLS estimator is also the maximum likelihood estimator when the error on d_{nm} is assumed to be white Gaussian noise.

1.2 Our approach: convex optimization

In this section, we give a brief introduction to convex optimization, which is the main tool we use to study the protein structural calculation problem. We first state a few definitions and properties of convex sets and functions. These are standard materials that can be found in many excellent convex optimization textbooks [120, 23, 14].

Definition 1.2.1. *A set $\mathcal{S} \in \mathbb{R}^d$ is convex if and only if for all $x, y \in \mathcal{S}$, $\theta x + (1-\theta)y \in \mathcal{S}$ for $\theta \in [0,1]$.*

Definition 1.2.2. *A function $f : \mathcal{S} \to \mathbb{R}$ is convex if and only if for all $x, y \in \mathcal{S}$, $f(\theta x + (1-\theta)y) \leq \theta f(x) + (1-\theta)f(y)$ for $\theta \in [0,1]$*

The importance of convex functions and convex sets is manifested in the following theorem.

Theorem 1.2.3. *For a convex function $f : \mathscr{S} \to \mathbb{R}$ where $\mathscr{S} \in \mathbb{R}^d$ is a convex set, any local minimizer of f is the global minimizer of f.*

Therefore to solve the optimization problem

$$\min_{x \in \mathscr{S}} f(x) \tag{1.15}$$

where f and $\mathscr{S}$ are both convex, it suffices to find a $x^* \in \mathscr{S}$ that minimizes $f(x)$ locally. This fact allows many instances of convex optimization problems to be solved efficiently. For example, in the unconstrained optimization problem (i.e. $\mathscr{S} = \mathbb{R}^d$) where f is convex and differentiable, the direct application of gradient descent yields the global minimizer of the optimization problem. Therefore in the field of mathematical programming, often convexity defines whether an optimization problem is easy or difficult. Rockafellar plainly states that "the great watershed in optimization isn't between linearity and nonlinearity, but convexity and nonconvexity" [119].

1.2.1 Semidefinite programming

One of the most studied examples of convex optimization are the conic optimization problem

$$\begin{aligned} & \min_{x \in \mathbb{R}^d} < c, x >_{\mathscr{K}} \\ \text{s.t.} \quad & < a_i, x >_{\mathscr{K}} \leq b_i, \ i = 1, \ldots, m, \\ & x \in \mathscr{K}, \end{aligned} \tag{1.16}$$

where $\mathscr{K}$ is a cone in $\mathbb{R}^d$ and $< \cdot, \cdot >_{\mathscr{K}}$ is the inner product specific to the cone $\mathscr{K}$ under consideration. This problem encapsulates several instances of convex optimization

problems that are solvable in polynomial time. A special case of conic optimization is linear programming (LP), where the cone considered is the non-negative orthant. In this thesis, we focus on the optimization problem over the cone of positive semidefinite matrices

$$\begin{aligned} &\min_{X\in\mathbb{R}^{d\times d}} \operatorname{Tr}(CX) \\ \text{s.t.} \quad &\operatorname{Tr}(A_i X) \leq b_i,\ i = 1,\dots,m, \\ &X \succeq 0, \end{aligned} \tag{1.17}$$

i.e. a semidefinite program. For LP and SDP, there exists polynomial time algorithm to solve the associated conic optimization problem to arbitrary precision. In 1979, Khachiyan proposed the polynomial time ellipsoid algorithm [88] to solve LP. Since the ellipsoid algorithm is impractical to use in practice, in 1984 Karmarkar proposed the more efficient interior point method [85] for LP, which paves a way to solving SDP in reasonable amount of time. Besides interior point methods, more recently, first order methods such as SDPLR [25] and alternating direction method of multiplier (ADMM) [22, 157] have been employed to solve SDP of large sizes. We note that not every conic optimization problem admits a polynomial time solutions. For example, optimization over the cone of co-positive matrices is NP-hard [46] due to the difficulty of membership testing.

1.2.2 Convex relaxation

Most problems encountered in realistic applications are non-convex. However, convex optimization provides a way to obtain the global optimizer of a non-convex problem via *convex relaxation*. In general, the idea of convex relaxation involves relaxing the domain of a non-convex problem into a larger set that is convex. If the new domain is close enough to the original domain, the solution to the new convex problem gives a

close approximation to the original problem. Moreover, if the solution to the convex-relaxed problem lies in the domain of the original problem, we are sure that such solution presents a minimizer to the original non-convex problem. This is in contrast to optimization techniques such as gradient descent or simulated annealing in that there is no way to know the optimality of the solution. In the following, we provide a few examples on the use of convex relaxation.

The bipartite graph matching problem [102] is one of the most well-known combinatorial optimization problem. A bipartite graph contains two partitions of nodes where weighted edges only exist between the two partitions. Let the weight $W(i,j)$ on an edge (i,j) describes the utility of pairing of nodes i,j. The bipartite graph matching problem can be defined as the combinatorial optimization problem

$$\max_{P \in \text{Perm}(n)} \text{Tr}(WP) \tag{1.18}$$

where $\text{Perm}(n)$ is the set of $n \times n$ permutation matrices. Intuitively, the solution to the optimization problem provides matchings between the two partitions that maximize the total utility. Despite the fact that the search space of this non-convex problem is combinatorially large, this problem can be solved in polynomial time via LP. The trick is to relax the domain of permutation matrices to the set of doubly stochastic matrices

$$\text{DS}(n) = \{P \mid P\mathbf{1} = \mathbf{1}, P^T\mathbf{1} = \mathbf{1}, P \geq 0\},$$

and solve the LP

$$\max_{P \in \text{DS}(n)} \text{Tr}(WP). \tag{1.19}$$

The Birkhoff-von Neumann theorem states that the set of doubly stochastic matrices is a polytope where the vertices of the polytope are the permutation matrices [102]. In other words, $\text{DS}(n)$ forms the convex hull of $\text{Perm}(n)$. Since the maximum of a generic

linear function in a polytope necessarily occurs on an extreme point of the polytope, the LP in (1.19) should always return a solution in Perm(n). This is a simple example that illustrates how a convex relaxation exactly recovers the solution to the original non-convex problem. This is enabled by a cost function with some nice structures, and a relaxed domain that is sufficiently close to the original non-convex domain.

The previous LP example provides solution to a problem that is solvable in polynomial time. Next we present a classic SDP relaxation to approximately solve a NP-hard combinatorial problem, the Max-Cut problem [56]. The Max-Cut problem attempts to find a partitioning of a graph such that the edges across the partitions are maximized. Now let $W \in \mathbb{R}^{n\times n}$ denotes the adjacency matrix of a graph. The Max-Cut problem can be formulated as

$$\max_{x\in\{\pm 1\}^n} \sum_{ij} \frac{1-x(i)x(j)}{2} W(i,j) \tag{1.20}$$

where $x(i) = 1$ ($x(i) = -1$) indicates that node i belongs to the first partition (second partition). From a complexity theory point of view, this problem is difficult due to its NP-hardness. From an optimization point of view, the difficulty is due to the search space that is non-convex and exponentially large (of size 2^n), rendering the search of global optimizer intractable. However, this problem admits an efficient SDP relaxation. Let

$$X = xx^T, \tag{1.21}$$

or equivalently,

$$X \succeq 0, \qquad \text{rank}(X) = 1, \tag{1.22}$$

the Max-Cut problem can be written as

$$\max_{X\succeq 0,\ \text{rank}(X)=1} \sum_{ij} \frac{1-X(i,j)}{2} W(i,j), \quad \text{s.t. } X(i,i) = 1. \tag{1.23}$$

While this is just an equivalent formulation of the original Max-Cut problem in terms

of the rank one matrix variable X, when the non-convex rank constraint is dropped, a convex relaxation

$$\max_{X\succeq 0} \sum_{ij} \frac{1-X(i,j)}{2} W(i,j), \quad \text{s.t. } X(i,i)=1. \tag{1.24}$$

is obtained. In general, the solution X^* needs not be rank one, hence being non-integral. However, a randomized rounding procedure is proposed [56] to round a high rank X^* to a rank one integer solution X_{round} with optimality guarantee. In particular,

$$\mathbb{E}_{X_{\text{round}}}\left[\sum_{ij} \frac{1-X_{\text{round}}(i,j)}{2} W(i,j)\right] \geq 0.878 \max_{x\in\{\pm 1\}^n} \sum_{ij} \frac{1-x(i)x(j)}{2} W(i,j). \tag{1.25}$$

where the expectation is taken over the randomly rounded solutions.

In many NP-hard problems, SDP relaxation can be used to identify polynomial-time solvable instances for these problems (see [27, 1, 151, 79] for example). A good example of such usage of SDP is the SNLSDP algorithm proposed to solve the distance geometry problem, which is described later in (1.31). For these SDP relaxations, conditions on the input data that ensure the convex relaxation has a unique solution in the original domain can often be analyzed mathematically. This will be illustrated more clearly in Chapter 2, where such uniqueness conditions are analyzed for the proposed convex relaxation for the global registration problem.

1.3 Review of current structural calculation methods

In this section, we give an overview of the methods used in protein structural calculation.

1.3.1 NOE-based methods

We first survey the methods that focus on solving the distance geometry problem in structural calculation. The distance geometry problem has a rather long history in the

field of applied mathematics and it is sometimes coined the name graph realization or graph embedding problem. It is known that if the full distance matrix D with entries $D(i,j) = \|x_i - x_j\|$ is available, the coordinates of the points can be obtained via spectral method. Let $X = [x_1, \ldots, x_K] \in \mathbb{R}^{3\times K}$, it can be verified that

$$X^T X = -\frac{1}{2} H(D \odot D)H \tag{1.26}$$

where

$$H = I_K - (1/K)\mathbf{1}\mathbf{1}^T \tag{1.27}$$

and $\odot$ is the Hadamard dot product. Therefore, given a distance matrix D the application of Cholesky factorization to $-(1/2)\ H(D \odot D)H$ results the desired coordinates. This method is known as classical multidimensional scaling. However, the problem becomes difficult in the presence of incomplete distance data. Indeed, [125] showed that the distance geometry problem is NP-hard. However, since NP-hard is a worst-case notion of computational hardness, it is possible that the average instances of the distance geometry problem can be solved efficiently.

The transformation between the Gram matrix $X^T X$ and $D \odot D$ in (1.26) gives rise to early algorithms to solve for protein structure using distance measurements. [71] constructs upper and lower bound to missing pairwise distances by applying triangle and tetra-angle inequalities to known pairwise distances. Then distance matrices D are then sampled according to the constructed bounds and CMDS is applied to check whether a rank 3 embedding is obtained. It is obvious that this is a brute-force search method and one might hope to improve the efficiency by optimization techniques. Local optimization based on steepest descent or majorize-minimization [20] thus have been applied to optimize the cost

$$\sum_{(i,j)\in E_{NOE}} (\|x_i - x_j\| - D(i,j))^2 \tag{1.28}$$

in order to find the coordinates. However, these methods are plagued with local minima issues.

In the field of biomolecular NMR, the mainstream approach to resolve the local minima issue is through simulated annealing [91, 69, 35, 128]. In simulated annealing, the "tunneling" mechanism pushes the solution out of a local minimum with probability $\exp(-\Delta E/T)$ where ΔE is the potential barrier. This procedure can be run for a long period of time in order to increase the chances of escaping local minima. In principle, this gives simulated annealing versatility to deal with arbitrary non-convex energy functions and such versatility is often favored by NMR practitioners. The main disadvantages of this type of methods are the speed and accuracy. Long cooling time may be required if the energy landscape is rugged. Moreover, due to the stochastic nature of the algorithm, even when the data is clean it is difficult to obtain a solution of high accuracy.

Within the last decade, the field of applied mathematics has witnessed a rapid development of using convex relaxation methods, in particular, semidefinite programming relaxation methods to solve non-convex problems with strong theoretical guarantees. The general idea of convex relaxation is to replace a non-convex problem with a convex surrogate problem such that the global optimums of these problems coincide. For the distance geometry problem, [137, 17] apply this technique to minimize

$$\sum_{(i,j)\in E_{NOE}} (\|x_i - x_j\|^2 - D(i,j)^2)^2 \tag{1.29}$$

to retrieve the 3D coordinates from the distances. To derive a convex relaxation, let $Y = X^T X$. Notice that since

$$Y = X^T X \quad \Longleftrightarrow \quad Y \succeq 0, \mathrm{rank}(Y) = 3, \tag{1.30}$$

the problem of finding an embedding from distances become

$$\begin{aligned} & \min_{Y} \sum_{(i,j)\in E_{NOE}} ((e_i - e_j)^T Y (e_i - e_j) - D(i,j)^2)^2 \\ \text{s.t.} \quad & Y \succeq 0 \\ & Y\mathbf{1} = 0 \\ & \text{rank}(Y) = 3 \end{aligned} \tag{1.31}$$

where the set of e_i's is the canonical basis in $\mathbb{R}^K$. The second constraint implies $X\mathbf{1} = 0$, meaning the coordinates are centered at zero. This optimization problem is non-convex due to the rank constraint on Y. A convex relaxation to this problem can be derived, if the rank-3 constraint is dropped, by paying the price of enlarging the search space from $3K$ number of variables (X) to K^2 number of variables (Y). Such technique of relaxing the rank constraint of a PSD variable will permeate this thesis. We note that this convex relaxation has a physical interpretation, that is to find an embedding in dimension K (instead of 3) that satisfies the distance constraints. By solving the distance geometry problem through such convex relaxation, a class of distance geometry problems that is solvable in polynomial time is identified. More precisely, if the given distance measurements only admit a unique embedding in any dimension [137, 61], the coordinates can be retrieved exactly via (1.31).

Though having a polynomial running time, SNLSDP can be expensive to use in practice. Thus a number of approaches attempt to speed up the running time. [153] proposes a further relaxation of SNLSDP by enforcing PSD constraints only on certain subblocks of Y. Furthermore, under an ADMM optimization framework the computation can be made parallelized [132]. [4] exploits the fact that in the presence of cliques with exact distance measurements, the range of Y is restricted and hence Y can be replaced by a smaller PSD variable. A separate line of research achieves speed up via divide-and-conquer approach. Almost all approaches of this type are variations

of the following theme: (1) Divide the points into overlapping patches, (2) embed the patches in $\mathbb{R}^3$, (3) stitch the patches coherently. This kind of approach has been applied successfully to the sensor network localization problem [169, 40], which is basically the distance geometry problem in 2D. For the problem of determining the molecular structure, while [41, 97] both use SNLSDP to embed the patches, there is a significant difference in how the patches are stitched together. As mentioned earlier, the solution to the distance geometry problem has a trivial rigid transformation ambiguity. This ambiguity issue becomes a problem when using a divide-and-conquer approach. After embedding each patch, the patch coordinates now have a rigid transformation ambiguity that needs to be determined in order to ensure the patches that share common points may be stitched coherently. [97] uses a sequential approach to build-up the global molecular structure from smaller patches. It leverages the fact that a closed-form solution can be obtained from singular value decomposition to determine the relative rigid transformation between two patches. However, greedy sequential approach allows error to accumulate, therefore [41] proposes a method to consider all the pairwise transformations at once via a notion of diffusion of the orthogonal matrices. Still the solution of [41] is not satisfactory, since the translations and orthogonal transformations of the patches are considered separately under such framework.

We remark that a lot of tools developed for solving the distance geometry problem also have applications in the field of sensor network localization (SNL). The SNL problem is an instance of the distance geometry problem in 2D. In some sense, the 2D distance geometry problem is simpler than its 3D version, in that there is combinatorial characterization on the uniqueness of solution [73]. Moreover, in the SNL problem, often there are anchor sensors that have known 2D coordinates. In many cases, it is also possible to obtain angular information for sensors within close proximity [114]. From the domain of SNL, the works most closely related to the approach presented in Chapter 2.6 are the local to global approaches [62, 169, 133, 40]. Again, these methods generally start by

identifying locally or globally rigid patches such that each patch is determined up to an affine or rigid transformation, then the transformations are determined such that relative relationship between the patches are preserved.

1.3.2 RDC-based methods

In [38], with RDC measured in two alignment media, a high resolution structure of the ubiquitin is obtained via a simulated annealing based method. This stirs a great interest in using RDC to obtain protein structures from solution NMR with resolution comparable to X-ray structures. Introducing the RDC potential term

$$\left(r_{nm} - \frac{(x_n - x_m)^T S (x_n - x_m)}{d_{nm}^2}\right)^2. \tag{1.32}$$

yields, however, a rugged energy landscape with sharp local minima that hinders the success of finding the correct conformation in the absence of a good initial structure [33, 11]. For example, [109] reports that direct minimization of the RDC potential using simulated annealing can yield structures that are as much as 20 Å away from the ground truth. When using simulated annealing, a popular approach to find the protein structure with RDC being the main constraints is through molecular fragment replacement (MFR) [92]. MFR finds homologous short fragments of the protein in the Protein Data Bank with the aid of RDC and chemical shifts. The fragments are then merged together to form an initial structure that will be locally refined by simulated annealing. However, using existing structures as initialization might lead to model bias. Moreover, there is still no guarantee that the initialization is good enough to avoid getting stuck at a local minima. For example, [109] reports that direct minimization of the RDC potential using simulated annealing can yield structures that are as much as 20 Å away from the ground truth.

Besides stochastic optimization, more recently a number of deterministic approaches

based on branch and prune [166, 30] and dynamic programming [109] have been proposed to find the globally optimal backbone structure. In particular, RDC-ANALYTIC [166, 150] exploits the fact that in the presence of two RDC measurements per amino-acid, the torsion angles that determine the orientation of an amino-acid have 16 possibilities, and a solution tree with a total of 16^M possible structures can be built for a protein with M amino-acids. The main advantage of branch and prune type methods is their ability to deal with sparse RDC datasets when used with an efficient pruning device such as the Ramachandran plot [116] (the empirical distribution of the torsion angles) and NOE. The dynamic programming approach [109] attempts to improve the robustness of the solution in tree searching based methods. However, as pointed out by the authors, it cannot readily incorporate additional information such as dihedral angles and distance restraints to improve the solution quality. Another approach with a similar flavor to the tree-searching based methods, REDCRAFT [24], performs Monte-Carlo sampling of the torsion angles of a protein according to Ramachandran plot. RDC measurements are then used to select the possible torsion angles. In general, the methods based on building a conformation space and pruning the unwanted conformations can lead to a relatively slow running time. Both REDCRAFT and RDC-Analytics need an hour or two to solve for the structure of typical size protein.

1.4 Main contributions

The main contribution in this thesis is to present fast and accurate algorithms for structural determination. In the first part of the thesis, we present a divide-and-conquer approach to solve the structural determination problem from distance restraints. To this end, in Chapter 2 we propose the Global Registration over Euclidean Transformation (GRET) method that stitches different parts of the protein coherently into a global structure. This chapter is based on the work published in [32]. Unlike previous

stitching methods that have been used to solve the distance geometry problem in a distributed fashion, our proposed method solves for both orthogonal transformations and translations for all protein fragments jointly under a maximum-likelihood estimation framework. We show that the non-convex problem of global registration surprisingly admits a Goemans-Williamson type Max-Cut SDP relaxation [56], allowing the problem to be solved globally and efficiently. We prove that despite the convex relaxation, we solve the global registration problem exactly when there is no noise in the data, and stably in the presence of noise. In Chapter 2.6, we present how the proposed global registration algorithm can be used to determine the structure of a protein and how various NMR data types can be considered. Moreover, in Chapter 4 we show that if given RDC data, the relative orientations of the Saupe tensors for each patch can be used to enhance the results of global registration. During this study we observe that whenever the structure of interest has large structural noise, the eigenvalues of the estimated Saupe tensor have magnitude systematically smaller than their actual values. This leads to systematic error when calculating the eigenvalue dependent parameters such as tensor magnitude and rhombicity. We then propose a Monte Carlo simulation method to remove such bias. We further demonstrate the effectiveness of our method in the setting of a divide-and-conquer approach, i.e. when the eigenvalue estimates from multiple template protein fragments are available and their average is used as an improved eigenvalue estimator. We note that this chapter is based on [89] which is a work to be submitted.

However, the divide-and-conquer procedure is essentially a distance based procedure in that it only uses distance restraints when embedding each patch in $\mathbb{R}^3$ before the stitching procedure. Thus it does not work well when the NOE restraint list is incomplete or ambiguous. Indeed, in our simulation, we are only able to determine the structure of the protein 1GB1 with an accuracy similar to the existing techniques. Therefore in Chapter 3, we propose a RDC-based method for embedding that alleviates the burden

of using NOE for protein structuring. This chapter is based on the submitted work [90]. We limit our attention to the calculation of protein backbone structure, leveraging the RDC and NOE measurements for the backbone. Unlike previous convex relaxation approaches that focused solely on distance constraints, we propose an SDP relaxation for backbone structure determination that simultaneously incorporates both NOE and RDC measurements. An additional advantage of this combination method is that it can provide accurate solutions even when using RDC alone. Our proposed SDP algorithm resolves the Open Problem posed in [45, Chapter 36]: *"Use SDP and the concept of distance geometry with angle restraints to model RDC-based structure determination."*. Our algorithmic contribution is that we provide a solution of the non-convex structural calculation problem by relaxing the search space to a set of positive semidefinite matrices (PSD). Numerically, our proposed methods recover the optimal solution exactly when there is no noise in the RDC, and stably when noise is added to the RDC. In some sense, the structural calculation problem from RDC measurements can be regarded as the distance geometry problem in a metric space (corresponding to the Saupe tensor) different from the standard Euclidean space. Since the convex relaxations in [137, 17] proposed for the distance geometry problem only involve the Gram matrix (inner product matrix) of the atom coordinates in the Euclidean space, these methods do not readily generalize to deal with RDC measurements that come from different inner product spaces. Such complication gives rise to the open problem in [45] and our idea is to use a convex relaxation that involves outer products of the atom coordinates to solve the distance geometry problem in multiple inner product spaces. We further exploit the fact that a protein backbone is better viewed as multiple rigid units that are chained together, rather than just a loose set of points. The coordinates of the atoms can thus be determined by the rotations of these rigid units. Our convex-relaxed optimization problem explicitly solves for the rotations of individual units *jointly* instead of the atom coordinates. This has the advantage of lowering the number of variables and allowing

facile incorporation of chirality constraints. Unlike existing optimization approaches in torsion angle space [69], with RDC measurements alone the cost and the constraints in our formulation are separable in the optimization variables (the rotations), i.e. each term in the cost and constraints only depends on a single rotation. This leads to an extremely efficient convex program- RDC-SDP with running time of about an order of magnitude faster than existing toolboxes that use RDC for *de novo* calculation of the protein backbone [24, 166]. This is rather remarkable as the computational problem of determining the orientations has its domain on the product manifold of special orthogonal matrices, with a search space that is non-convex and exponential in size. Fast and accurate determination of the initial structure could have potential applications in quick validation of backbone and NOE resonance assignment [67, 167] or refining Saupe tensor estimate through alternating minimization. To include both RDC and NOE restraints, we propose a different SDP - RDC-NOE-SDP, at the expense of increasing the running time. We also tested the algorithms in calculating the structure of ubiquitin fragments from experimental RDC and NOE data deposited on the Protein Data Bank (PDB). We successfully computed the backbone structure for short fragments of ubiquitin (each consisting of 12 amino acids on average) up to 0.6 Å resolution. To further assess the quality of our structural calculation procedure, we introduce a classical statistical tool, the Cramér-Rao bound (CRB), which provides the minimum possible variance of the estimated atomic coordinates for a given noise model on the RDC and NOE. For most of the noise levels, our methods can achieve the CRB, even in the presence of only RDC measurements.

Chapter 2

Global registration over Euclidean transforms (GRET) and distributed protein structuring from NOE

The problem of point-cloud registration comes up naturally in distributed approaches to molecular conformation [51, 42], and also in computer vision and graphics [129, 145, 159]. The registration problem in question is one of determining the coordinates of a point cloud P from the knowledge of (possibly noisy) coordinates of smaller point cloud subsets (called *patches*) $P_1, \ldots, P_M$ that are derived from P through some general transformation. In this chapter, we consider the problem of *rigid registration* in which the points within a given P_i are (ideally) obtained from P through an unknown rigid transform. In some other applications [108, 145, 94], one is often interested in finding the optimal transforms (one for each patch) that consistently align $P_1, \ldots, P_M$. We note that this can be seen as a sub-problem in the determination of the coordinates of P [40, 121].

2.0.1 Two-patch registration

The particular problem of two-patch registration has been well-studied [52, 76, 8]. In the noiseless setting, we are given two point clouds $\{x_1, \ldots, x_K\}$ and $\{y_1, \ldots, y_K\}$ in $\mathbb{R}^d$, where the latter is obtained through some rigid transform of the former. Namely,

$$y_k = Ox_k + t \qquad (k = 1, \ldots, K), \tag{2.1}$$

where O is some unknown $d \times d$ orthogonal matrix (that satisfies $O^T O = I_d$) and $t \in \mathbb{R}^d$ is some unknown translation.

The problem is to infer O and t from the above equations. To uniquely determine O and t, one must have at least $K \geq d + 1$ non-degenerate points[1]. In this case, O can be determined simply by fixing the first equation in (2.1) and subtracting (to eliminate t) any of the remaining d equations from it. Say, we subtract the next d equations:

$$[y_2 - y_1 \ \cdots \ y_{d+1} - y_1] = O[x_2 - x_1 \ \cdots \ x_{d+1} - x_1].$$

By the non-degeneracy assumption, the matrix on the right of O is invertible, and this gives us O. Plugging O into any of the equations in (2.1), we get t.

In practical settings, (2.1) would hold only approximately, say, due to noise or model imperfections. A particular approach then would be to determine the optimal O and t by considering the following least-squares program:

$$\min_{O \in \mathbb{O}(d),\ t \in \mathbb{R}^d} \quad \sum_{k=1}^{K} \|y_k - Ox_k - t\|_2^2. \tag{2.2}$$

Note that the problem looks difficult a priori since the domain of optimization is $\mathbb{O}(d) \times \mathbb{R}^d$, which is non-convex. Remarkably, the global minimizer of this non-convex problem can be found exactly, and has a simple closed-form expression [50, 87, 74, 52, 76, 8].

[1] By non-degenerate, we mean that the affine span of the points is d dimensional.

More precisely, the optimal $O^\star$ is given by VU^T, where $U\Sigma V^T$ is the singular value decomposition (SVD) of

$$\sum_{k=1}^{K}(x_k - x_c)(y_k - y_c)^T,$$

in which $x_c = (x_1 + \cdots + x_K)/K$ and $y_c = (y_1 + \cdots + y_K)/K$ are the centroids of the respective point clouds. The optimal translation is $t^\star = y_c - O^\star x_c$.

The fact that two-patch registration has a closed-form solution is used in the so-called incremental (sequential) approaches for registering multiple patches [15]. The most well-known method is the ICP algorithm [121] (note that ICP uses other heuristics and refinements besides registering corresponding points). Roughly, the idea in sequential registration is to register two overlapping patches at a time, and then integrate the estimated pairwise transforms using some means. The integration can be achieved either locally (on a patch-by-patch basis), or using global cycle-based methods such as synchronization [129, 77, 134, 145, 152]. More recently, it was demonstrated that, by locally registering overlapping patches and then integrating the pairwise transforms using synchronization, one can design efficient and robust methods for distributed sensor network localization [40] and molecular conformation [42]. Note that, while the registration phase is local, the synchronization method integrates the local transforms in a globally consistent manner. This makes it robust to error propagation that often plague local integration methods [77, 152].

2.0.2 Multi-patch registration

To describe the multi-patch registration problem, we first introduce some notations. Suppose $x_1, x_2, \ldots, x_K$ are the unknown global coordinates of a point cloud in $\mathbb{R}^d$. The point cloud is divided into patches $P_1, P_2, \ldots, P_M$, where each P_i is a subset of $\{x_1, x_2, \ldots, x_K\}$. The patches are in general overlapping, whereby a given point can belong to multiple patches. We represent this membership using an undirected bipartite

graph $\Gamma = (V_x \cup V_P, E)$. The set of vertices $V_x = \{x_1, \ldots, x_K\}$ represents the point cloud, while $V_P = \{P_1, \ldots, P_M\}$ represents the patches. The edge set $E = E(\Gamma)$ connects V_x and V_P, and is given by the requirement that $(k, i) \in E$ if and only if $x_k \in P_i$. We will henceforth refer to Γ as the *membership graph*.

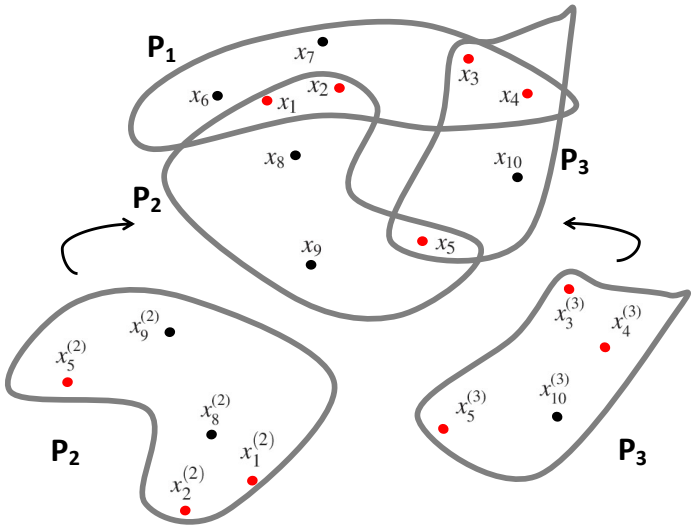


Figure 2.1: The problem of registering 3 patches on $\mathbb{R}^2$. One is required to find the global coordinates of the points from the corresponding local patch coordinates. The local coordinates of the points in patches P_2 and P_3 are shown (see (2.5) for the notation of local coordinates). It is only the common points (belonging to two or more patches, marked in red) that contribute to the registration. Note that sequential or pairwise registration would fail in this case. This is because no pair of patches can be registered as they have less than 3 points in common (at least 3 points are required to fix rotations, reflections, and translations in $\mathbb{R}^2$). The proposed SDP-based algorithm proposed does a global registration, and is able to recover the exact global coordinates for this example.

We assume that the local coordinates of a given patch can (ideally) be related to the global coordinates through a single rigid transform, that is, through some rotation, reflection, and translation. More precisely, with every patch P_i we associate some (unknown) orthogonal transform O_i and translation t_i. If point x_k belongs to patch P_i, then its representation in P_i is given by (cf. (2.1) and Figure 2.1)

$$x_k^{(i)} = O_i^T (x_k - t_i) \qquad (k, i) \in E(\Gamma). \tag{2.3}$$

Alternatively, if we fix a particular patch P_i, then for every point belonging to that patch,

$$x_k = O_i x_k^{(i)} + t_i \qquad (k,i) \in E(\Gamma). \tag{2.4}$$

In particular, a given point can belong to multiple patches, and will have a different representation in the coordinate system of each patch.

We assume that we are given the membership graph and the local coordinates (referred to as measurements), namely

$$\Gamma \quad \text{and} \quad \{x_k^{(i)}, (k,i) \in E(\Gamma)\}, \tag{2.5}$$

and the goal is to recover the coordinates $x_1, \ldots, x_K$, and in the process the unknown rigid transforms $(O_1, t_1), \ldots, (O_M, t_M)$, from (2.5). Note that the global coordinates are determined up to a global rotation, reflection, and translation. We say that two points clouds (also called *configurations*) are *congruent* if one is obtained through a rigid transformation of the other. We will always identify two congruent configurations as being a single configuration.

Under appropriate non-degeneracy assumptions on the measurements, one task would be to specify appropriate conditions on Γ under which the global coordinates can be uniquely determined. Intuitively, it is clear that the patches must have enough points in common for the registration problem to have an unique solution. For example, it is clear that the global coordinates cannot be uniquely recovered if Γ is disconnected.

In practical applications, we are confronted with noisy settings where (2.4) holds only approximately. In such cases, we would like to determine the global coordinates and the rigid transforms such that the discrepancy in (2.4) is minimal. In particular, we

consider the following quadratic loss:

$$\phi = \sum_{(k,i)\in E(\Gamma)} \|x_k - O_i x_k^{(i)} - t_i\|^2, \tag{2.6}$$

where $\|\cdot\|$ is the Euclidean norm on $\mathbb{R}^d$. The optimization problem is to minimize ϕ with respect to the following variables:

$$x_1, x_2, \ldots, x_K \in \mathbb{R}^d, \quad O_1, \ldots, O_M \in \mathbb{O}(d), \quad t_1, \ldots, t_M \in \mathbb{R}^d.$$

The input to the problem are the measurements in (2.5). Note that our ultimate goal is to determine $x_1, x_2, \ldots, x_K$; the rigid transforms can be seen as latent variables.

The problem of multipatch registration is intrinsically non-convex since one is required to optimize over the non-convex domain of orthogonal transforms. Different ideas from the optimization literature have been deployed to attack this problem, including Lagrangian optimization and projection methods. In the Lagrangian setup, the orthogonality constraints are incorporated into the objective; in the projection method, the constraints are forced after every step of the optimization [115]. Following the observation that the registration problem can be viewed as an optimization on the Grassmanian and Stiefel manifolds, researchers have proposed algorithms using ideas from the theory and practice of manifold optimization [94]. A detailed review of these methods is beyond the scope of this chapter, and instead we refer the interested reader to these excellent reviews [47, 2]. Manifold-based methods are, however, local in nature, and are not guaranteed to find the global minimizer. Moreover, it is rather difficult to certify the noise stability of such methods.

2.0.3 Contributions

The main contributions in this chapter can be organized into the following categories:

1. **Algorithm**: In Section 2.1, we demonstrate that the above least-squares problem can be reduced to the following subproblem:

$$\max_{O_1,\ldots,O_M} \sum_{i,j=1}^{M} \mathrm{Tr}(O_i C_{ij} O_j^T) \quad \text{subject to} \quad O_1,\ldots,O_M \in \mathbb{O}(d), \tag{2.7}$$

where the block matrix $C \in \mathbb{R}^{Md\times Md}$ is positive semidefinite, and where we use $C_{ij} \in \mathbb{R}^{d\times d}(1 \le i,j \le M)$ to denote the (i,j)-th block of C. Given the solution of (2.7), the desired global coordinates can simply be obtained through a linear transformation of the solution.

Next, we observe that (2.7) can be relaxed into a tractable convex program, namely a semidefinite program. This yields a tractable algorithm for global registration that is described in Algorithm 2. The corresponding algorithm derived from the spectral relaxation of (2.7), that was already considered in [94, 168, 59], is described in Algorithm 1 for reference.

2. **Exact Recovery**: In Section 2.2, we study conditions on the coefficient matrix C in (2.7) for exact recovery using Algorithms 1 and 2. In particular, we show that the exact recovery questions Algorithms 2 about can be mapped into rigidity theoretic questions that have already been investigated earlier. in [168, 59]. The contribution of this section is the connection made between the C matrix in (2.7) and various prior notions of rigidity considered in [168, 59]. In Section 2.3, we present an efficient randomized rank test for C than can be used to certify exact recovery (motivated by the work in [73, 60, 135]).

3. **Stability Analysis**: In Section 2.4, we study the stability of Algorithms 1 and 2 for the noise model in which the patch coordinates are perturbed using noise of

bounded size (note that the stability of the spectral relaxation was not investigated in [94]). Our main result here is Theorem 2.4.2 which states that, if C has a particular rank, then the registration error for the semidefinite relaxation is within a constant factor of the noise level. To the best of our knowledge, there is no existing algorithm for multipatch registration that comes with a similar stability guarantee.

4. **Application**: In Section 2.5, we present numerical results on simulated data to numerically verify the exact recovery and noise stability properties of Algorithms 1 and 2. Our main empirical findings are the following:
   (a) The semidefinite relaxation performs significantly better than spectral and manifold-based optimization (say, with the spectral solution as initialization) in terms of the reconstruction quality (see first plot in Figure 2.7).
   (b) The relaxation gap is mostly zero for the semidefinite program (we are able to solve the original non-convex problem) up to a certain noise threshold (see second plot in Figure 2.7).
   In Section 2.6, we apply our algorithm to determine the protein structure from both simulated and experimental data in a distributed fashion.

2.0.4 Broader context and related work

The objective (2.6) is a straightforward extension of the objective for two-patches [50, 52, 76, 8]. In fact, this objective was earlier considered by Zhang et al. for distributed sensor localization [169]. The present work is also closely tied to the work of Cucuringu et al. on distributed localization [40, 42], where a similar objective is implicitly optimized. The common theme in these works is that some form of optimization is used to globally register the patches, once their local coordinates have been determined by some means.

There is, however, some fundamental differences between the various algorithms used to actually perform the optimization. Zhang et al. [169] use alternating least-squares to iteratively optimize over the global coordinates and the transforms, which to the best of our knowledge has no convergence guarantee. On the other hand, Cucuringu et al. [40, 42] first optimize over the orthogonal transforms (using synchronization [134]), and then solve for the translations (in effect, the global coordinates) using least-squares fitting. In this work, we combine these different ideas into a single framework. While our objective is similar to the one used in [169], we jointly optimize the rigid transforms and positions. In particular, the algorithms considered in Section 2.1 avoid the convergence issues associated with alternating least-squares in [169], and is able to register patch systems that cannot be registered using the approach in [40, 42].

Another closely related work is the paper by Krishnan et al. on global registration [94], where the optimal transforms (rotations to be specific) are computed by extending the objective in (2.2) to the multipatch case. The subsequent mathematical formulation has strong resemblance with our formulation, and, in fact, leads to a subproblem that is equivalent to (2.7). Krishnan et al. [94] propose the use of manifold optimization to solve (2.7), where the manifold is the product manifold of rotations. However, as mentioned earlier, manifold methods generally do not offer guarantees on convergence (to the global minimum) and stability. Moreover, the manifold in (2.7) is not connected. Therefore, any local method cannot solve (2.7) if the initial guess is on the wrong component of the manifold.

It is exactly at this point that we depart from [94], namely, we propose to relax (2.7) into a tractable semidefinite program (SDP). This was motivated by a long line of work on the use of SDP relaxations for non-convex (particularly NP-hard) problems. See, for example, [103, 56, 162, 111, 26, 96], and these reviews [146, 112, 170]. Note that for $d = 1$, (2.7) is a quadratic Boolean optimization, similar to the `MAX-CUT` problem. An SDP-based algorithm with randomized rounding for solving `MAX-CUT` was proposed in

the seminal work of Goemans and Williamson [56]. The semidefinite relaxation that we consider in Section 2.1 is motivated by this work. In connection with the present work, we note that provably stable SDP algorithms have been considered for low rank matrix completion [26], phase retrieval [27, 149], and graph localization [81].

We note that a special case of the registration problem considered here is the so-called generalized Procrustes problem [63]. Within the point-patch framework just introduced, the goal in Procrustes analysis is to find $O_1, \dots, O_M \in \mathbb{O}(d)$ that minimizes

$$\sum_{k=1}^{K} \sum_{i,j=1}^{M} \|O_i x_k^{(i)} - O_j x_k^{(j)}\|^2. \tag{2.8}$$

In other words, the goal is to achieve the best possible alignment of the M patches through orthogonal transforms. This can be seen as an instance of the global registration problem without the translations ($t_1 = \cdots = t_M = 0$), and in which Γ is complete. It is not difficult to see that (2.8) can be reduced to (2.7). On the other hand, using the analysis in Section 2.1, it can be shown that (2.6) is equivalent to (2.8) in this case. While the Procrustes problem is known to be NP-hard, several polynomial-time approximations with guarantees have been proposed. In particular, SDP relaxations of (2.8) have been considered in [111, 136, 110], and more recently in [10]. We use the relaxation of (2.7) considered in [10] for reasons to be made precise in Section 2.1.

2.0.5 Notations

We summarize the notations used in this chapter. We will frequently use block matrices built from smaller matrices of size $d \times d$, where d is the dimension of the ambient space. For some block matrix A, we will use A_{ij} to denote its (i, j)-th block. We use $\mathbb{O}(d)$ to denote the group of orthogonal transforms (matrices) acting on $\mathbb{R}^d$, and $\mathbb{O}(d)^M$ to denote the M-fold product of $\mathbb{O}(d)$ with itself. We will also conveniently identify the matrix $[O_1 \cdots O_M]$ with an element of $\mathbb{O}(d)^M$ where each $O_i \in \mathbb{O}(d)$. We use e_i^N denotes

the all-zero vector of length N with 1 at the i-th position.

2.1 Spectral and semidefinite relaxations (GRET-SPEC and GRET-SDP)

The minimization of (2.6) involves unconstrained variables (global coordinates and patch translations) and constrained variables (the orthogonal transformations). We first solve for the unconstrained variables in terms of the unknown orthogonal transformations, representing the former as linear combinations of the latter. This reduces (2.6) to a quadratic optimization problem over the orthogonal transforms of the form (2.7).

In particular, we combine the global coordinates and the translations into a single matrix:

$$Z = \begin{bmatrix} x_1 & \cdots & x_K & t_1 & \cdots & t_M \end{bmatrix} \in \mathbb{R}^{d\times(K+M)}. \tag{2.9}$$

Similarly, we combine the orthogonal transforms into a single matrix,

$$O = [O_1 \ \cdots \ O_M] \in \mathbb{R}^{d\times Md}. \tag{2.10}$$

Recall that we will conveniently identify O with an element of $\mathbb{O}(d)^M$.

To express (2.6) in terms of Z and O, we write $x_k - t_i = Ze_{ki}$, where

$$e_{ki} = e_k^{K+M} - e_{K+i}^{K+M}.$$

Similarly, we write $O_i = O(e_i^M \otimes I_d)$. This gives us

$$\phi(Z,O) = \sum_{(k,i)\in E(\Gamma)} \|Ze_{ki} - O(e_i^M \otimes I_d)x_k^{(i)}\|^2.$$

Using $\|x\|^2 = \mathrm{Tr}(xx^T)$, and properties of the trace, we obtain

$$\phi(Z,O) = \mathrm{Tr}\left(\begin{bmatrix} Z & O \end{bmatrix} \begin{bmatrix} L & -B^T \\ -B & D \end{bmatrix} \begin{bmatrix} Z^T \\ O^T \end{bmatrix} \right), \tag{2.11}$$

where

$$\begin{aligned} L &= \sum_{(k,i)\in E} e_{ki} e_{ki}^T, \quad B = \sum_{(k,i)\in E} (e_i^M \otimes I_d) x_{k,i} e_{ki}^T, \text{ and} \\ D &= \sum_{(k,i)\in E} (e_i^M \otimes I_d) x_{k,i} {x_{k,i}}^T (e_i^M \otimes I_d)^T. \end{aligned} \tag{2.12}$$

The matrix L is the combinatorial graph Laplacian of Γ [34], and is of size $(K+M) \times (K+M)$. The matrix B is of size $Md \times (K+M)$, and the size of the block diagonal matrix D is $Md \times Md$.

The optimization program now reads

$$\text{(P)} \qquad \min_{Z,O} \ \phi(Z,O) \quad \text{subject to} \quad Z \in \mathbb{R}^{d\times(K+M)}, \ O \in \mathbb{O}(d)^M.$$

The fact that $\mathbb{O}(d)^M$ is non-convex makes (P) non-convex. In the next few Sections, we will show how this non-convex program can be approximated by tractable spectral and convex programs.

2.1.1 Optimization over translations

Note that we can write (P) as

$$\min_{O\in\mathbb{O}(d)^M} \Big[\min_{Z\in\mathbb{R}^{d\times(K+M)}} \ \phi(Z,O) \Big].$$

That is, we first minimize over the free variable Z for some fixed $O \in \mathbb{O}(d)^M$, and then we minimize with respect to O.

Fix some arbitrary $O \in \mathbb{O}(d)^M$, and set $\psi(Z) = \phi(Z, O)$. It is clear from (2.11) that $\psi(Z)$ is quadratic in Z. In particular, the stationary points $Z^\star = Z^\star(O)$ of $\psi(Z)$ satisfy

$$\nabla\psi(Z^\star) = 0 \quad \Rightarrow \quad Z^\star L = OB. \tag{2.13}$$

Note that the Hessian of $\psi(Z)$ equals $2L$, and it is clear from (2.12) that $L \succeq 0$. Therefore, $Z^\star$ is a minimizer of $\psi(Z)$.

If Γ is connected, then e is the only vector in the null space of L [34]. Let $L^\dagger$ be the Moore-Penrose pseudo-inverse of L, which is again positive semidefinite. It can be verified that

$$LL^\dagger = L^\dagger L = I_{K+M} - (K+M)^{-1}\mathbf{1}\mathbf{1}^T. \tag{2.14}$$

If we right multiply (2.13) by $L^\dagger$, we get

$$Z^\star = OBL^\dagger + t\mathbf{1}^T, \tag{2.15}$$

where $t \in \mathbb{R}^d$ is some global translation. Conversely, if we right multiply (2.15) by L and use the facts that $\mathbf{1}^T L = 0$ and $B\mathbf{1} = 0$, we get (2.13). Thus, every solution of (2.13) is of the form (2.15).

Substituting (2.15) into (2.11), we get

$$\psi(Z^\star) = \phi(Z^\star, O) = \mathrm{Tr}(CO^TO) = \sum_{i,j=1}^{M} \mathrm{Tr}(O_i C_{ij} O_j^T), \tag{2.16}$$

where

$$C = \begin{bmatrix} BL^\dagger & I_{Md} \end{bmatrix} \begin{bmatrix} L & -B^T \\ -B & D \end{bmatrix} \begin{bmatrix} L^\dagger B^T \\ I_{Md} \end{bmatrix} = D - BL^\dagger B^T. \tag{2.17}$$

Note that (2.16) has the global translation t taken out. This is not a surprise since ϕ is invariant to global translations. Moreover, note that we have not forced the orthogonal constraints on O as yet. Since $\phi(Z, O) \geq 0$ for any Z and O, it necessarily follows

from (2.16) that $C \succeq 0$. We will see in the sequel how the spectrum of C dictates the performance of the convex relaxation of (2.16).

In analogy with the notion of stress in rigidity theory [60], we can consider (2.6) as a sum of the "stress" between pairs of patches when we try to register them using rigid transforms. In particular, the (i, j)-th term in (2.16) can be regarded as the stress between the (centered) i-th and j-th patches generated by the orthogonal transforms. Keeping this analogy in mind, we will henceforth refer to C as the *patch-stress matrix*.

2.1.2 Optimization over orthogonal transforms

The goal now is to optimize (2.16) with respect to the orthogonal transforms, that is, we have reduced (P) to the following problem:

$$(\mathrm{P}_0) \qquad \min_{O \in \mathbb{R}^{d \times Md}} \operatorname{Tr}(CO^TO) \quad \text{subject to} \quad (O^TO)_{ii} = I_d \ (1 \leq i \leq M).$$

This is a non-convex problem since O lives on a non-convex (disconnected) manifold [2]. We will generally refer to any method which uses manifold optimization to solve (P_0) and then computes the coordinates using (2.15) as "Global Registration over Euclidean Transforms using Manifold Optimization" (`GRET-MANOPT`).

2.1.3 Spectral relaxation and rounding

Following the quadratic nature of the objective in (P_0), it is possible to relax it into a spectral problem. More precisely, consider the domain

$$\mathscr{S} = \{O \in \mathbb{R}^{d \times Md} \ : \ \text{rows of } O \text{ are orthogonal and each row has norm } \sqrt{M}\}.$$

That is, we do not require the $d \times d$ blocks in $O \in \mathscr{S}$ to be orthogonal. Instead, we only require the rows of O to form an orthogonal system, and each row to have the same

norm. It is clear that $\mathscr{S}$ is a larger domain than that determined by the constraints in (P_0). In particular, we consider the following relaxation of (P_0):

$$(P_1) \qquad \min_{O \in \mathscr{S}} \mathrm{Tr}(CO^T O).$$

This is precisely a spectral problem in that the global minimizers are determined from the spectral decomposition of C. More precisely, let $\mu_1 \leq \ldots \leq \mu_{Md}$ be eigenvalues of C, and let $r_1, \ldots, r_{Md}$ be the corresponding eigenvectors. Define

$$W^\star \stackrel{\text{def}}{=} \sqrt{M}\big[r_1 \cdots r_d\big]^T \in \mathbb{R}^{d \times Md}. \tag{2.18}$$

Then

$$\mathrm{Tr}(CW^{\star T} W^\star) = \min_{O \in \mathscr{S}} \mathrm{Tr}(CO^T O) = M(\mu_1 + \cdots + \mu_d). \tag{2.19}$$

Due to the relaxation, the blocks of $W^\star$ are not guaranteed to be in $\mathbb{O}(d)$. We round each $d \times d$ block of $W^\star$ to its "closest" orthogonal matrix. More precisely, let $W^\star = [W_1^\star \cdots W_M^\star]$. For every $1 \leq i \leq M$, we find $O_i^\star \in \mathbb{O}(d)$ such that

$$\|O_i^\star - W_i^\star\|_F = \min_{O \in \mathbb{O}(d)} \|O - W_i^\star\|_F.$$

As noted earlier, this has a closed-form solution, namely $O_i^\star = UV^T$, where $U\Sigma V^T$ is the SVD of $W_i^\star$. We now put the rounded blocks back into place and define

$$O^\star \stackrel{\text{def}}{=} \big[O_1^\star \ldots O_M^\star\big] \in \mathbb{O}(d)^M. \tag{2.20}$$

In the final step, following (2.15), we define

$$Z^\star \stackrel{\text{def}}{=} O^\star B L^\dagger \in \mathbb{R}^{d \times (K+M)}. \tag{2.21}$$

The first K columns of $Z^\star$ are taken to be the reconstructed global coordinates.

We will refer to this spectral method as the "Global Registration over Euclidean Transforms using Spectral Relaxation" (GRET-SPEC). The main steps of GRET-SPEC are summarized in Algorithm 1. We note that a similar spectral algorithm was proposed for angular synchronization by Bandeira et al. [9], and by Krishnan et al. [94] for initializing the manifold optimization.

Algorithm 1 GRET-SPEC

Require: Membership graph Γ, local coordinates $\{x_k^{(i)}, (k,i) \in E(\Gamma)\}$, dimension d.
Ensure: Global coordinates $x_1, \ldots, x_K$ in $\mathbb{R}^d$.
1: Build B, L and D in (2.12) using Γ.
2: Compute $L^\dagger$ and $C = D - BL^\dagger B^T$.
3: Compute bottom d eigenvectors of C, and set $W^\star$ as in (2.18).
4: **for** $i = 1$ to M **do**
5: **if** $W_i^\star \in \mathbb{O}(d)$ **then**
6: $O_i^\star \leftarrow W_i^\star$.
7: **else**
8: Compute $W_i^\star = U_i \Sigma_i V_i^T$.
9: $O_i^\star \leftarrow U_i V_i^T$.
10: **end if**
11: **end for**
12: $O^\star \leftarrow \left[O_1^\star \cdots O_M^\star\right]$
13: $Z^\star \leftarrow O^\star B L^\dagger$.
14: Return first K columns of $Z^\star$.

The question at this point is how are the quantities $O^\star$ and $Z^\star$ obtained from GRET-SPEC related to the original problem (P)? Since (P_1) is obtained by relaxing the block-orthogonality constraint in (P_0), it is clear that if the blocks of $W^\star$ are orthogonal, then $O^\star$ and $Z^\star$ are solutions of (P), that is,

$$\phi(Z^\star, O^\star) \leq \phi(Z, O) \qquad \text{for all } Z \in \mathbb{R}^{d\times(K+M)},\ O \in \mathbb{O}(d)^M.$$

We have actually found the global minimizer of the original non-convex problem (P) in this case.

Observation 2.1.1 (Tight relaxation using GRET-SPEC)**.** *If the $d \times d$ blocks of the solution*

of (P_1) *are orthogonal, then the coordinates and transforms computed by GRET-SPEC are the global minimizers of* (P).

If some the blocks are not orthogonal, the rounded quantities $O^\star$ and $Z^\star$ are only an approximation of the solution of (P).

2.1.4 Semidefinite relaxation and rounding

We now explain how we can obtain a tighter relaxation of (P_0) using a semidefinite program, for which the global minimizer can be computed efficiently. Our semidefinite program was motivated by the line of works on the semidefinite relaxation of non-convex problems [103, 56, 146, 26].

Consider the domain

$$\mathscr{C} = \{O \in \mathbb{R}^{Md \times Md} \ : \ (O^T O)_{11} = \cdots = (O^T O)_{MM} = I_d\}.$$

That is, while we require the columns of each $Md \times d$ block of $O \in \mathscr{C}$ to be orthogonal, we do not force the non-convex rank constraint $\mathrm{rank}(O) = d$. This gives us the following relaxation

$$\min_{O \in \mathscr{C}} \mathrm{Tr}(C O^T O). \tag{2.22}$$

Introducing the variable $G = O^T O$, (2.22) is equivalent to

$$(\mathrm{P}_2) \qquad \min_{G \in \mathbb{R}^{Md \times Md}} \mathrm{Tr}(CG) \quad \text{subject to} \quad G \succeq 0,\ G_{ii} = I_d \ (1 \le i \le M).$$

This is a standard semidefinite program [146] which can be solved using software packages such as `SDPT3` [141] and CVX [64]. We provide details about SDP solvers and their computational complexity later in Section 2.1.5.

Let us denote the solution of (P_2) by $G^\star$, that is,

$$\mathrm{Tr}(CG^\star) = \min_{G \in \mathbb{R}^{Md \times Md}} \{\mathrm{Tr}(CG) : G \succeq 0,\ G_{11} = \cdots = G_{MM} = I_d\}. \tag{2.23}$$

By the linear constraints in (P_2), it follows that $\mathrm{rank}(G^\star) \geq d$. If $\mathrm{rank}(G^\star) > d$, we need to round (approximate) it by a rank-d matrix. That is, we need to project it onto the domain of (P_0). One possibility would be to use random rounding that come with approximation guarantees; for example, see [56, 10]. In this work, we use deterministic rounding, namely the eigenvector rounding which retains the top d eigenvalues and discards the remaining. In particular, let $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{Md}$ be the eigenvalues of $G^\star$, and $q_1, \ldots, q_{Md}$ be the corresponding eigenvectors. Let

$$W^\star \stackrel{\text{def}}{=} \left[\sqrt{\lambda_1} q_1 \ \cdots \ \sqrt{\lambda_d} q_d\right]^T \in \mathbb{R}^{d \times Md}. \tag{2.24}$$

We now proceed as in the GRET-SPEC, namely, we define $O^\star$ and $Z^\star$ from $W^\star$ as in (2.20) and (2.21). We refer to the complete algorithm as "Global Registration over Euclidean Transforms using SDP" (GRET-SDP). The main steps of GRET-SDP are summarized in Algorithm 2.

Similar to Observation 2.1.1, we note the following for GRET-SDP.

Observation 2.1.2 (Tight relaxation using GRET-SDP)**.** *If the rank of the solution of* (P_2) *is exactly d, then the coordinates and transforms computed by GRET-SDP are the global minimizers of* (P).

If $\mathrm{rank}(G^\star) > d$, the output of GRET-SDP can only be considered as an approximation of the solution of (P). The quality of the approximation for (P_2) can be quantified using, for example, the randomized rounding in [10]. More precisely, note that since D is

Algorithm 2 GRET-SDP

Require: Membership graph Γ, local coordinates $\{x_k^{(i)}, (k,i) \in E(\Gamma)\}$, dimension d.
Ensure: Global coordinates $x_1, \ldots, x_K$ in $\mathbb{R}^d$.

```
Build B, L and D in (2.12) using Γ.
Compute L† and C = D − BL†B^T.
G* ← Solve the SDP (P_2) using C.
Compute top d eigenvectors of G*, and set W* using (2.24).
if rank(G*) = d then
    O* ← W*.
else
    for i = 1 to M do
        Compute W_i* = U_i Σ_i V_i^T.
        O_i* ← U_i V_i^T.
    end for
    O* ← [O_1* ··· O_M*]
end if
Z* ← O*BL†.
Return first K columns of Z*.
```

block-diagonal, (2.22) is equivalent (up to a constant term) to

$$\max_{O \in \mathscr{G}} \operatorname{Tr}(QO^TO)$$

where $Q = BL^\dagger B^T \succeq 0$. Bandeira et al. [10] show that the orthogonal transforms (which we continue to denote by $O^\star$) obtained by a certain random rounding of $G^\star$ satisfy

$$\mathbb{E}\left[\operatorname{Tr}(Q\, O^{\star T} O^\star)\right] \geq \alpha_d^2 \cdot \mathrm{OPT},$$

where OPT is the optimum of the unrelaxed problem (2.7) with $Q = BL^\dagger B^T$, and α_d is the expected average of the singular values of a $d \times d$ random matrix with entries iid $\mathcal{N}(0, 1/d)$. It was conjectured in [10] that α_d is monotonically increasing, and the boundary values were computed to be $\alpha_1 = \sqrt{2/\pi}$ (α_1 was also reported here [112]) and $\alpha_\infty = 8/3\pi$. We refer the reader to [10] for further details on the rounding procedure, and its relation to previous work in terms of the approximation ratio. Empirical results, however, suggest that the difference between deterministic and randomized rounding is

small as far as the final reconstruction is concerned. We will therefore simply use the deterministic rounding.

2.1.5 Computational complexity

The main computations in GRET-SPEC are the Laplacian inversion, the eigenvector computation, and the orthogonal rounding. The cost of inverting L when Γ is dense is $O((K+M)^3)$. However, for most practical applications, we expect Γ to be sparse since every point would typically be contained in a small number of patches. In this case, it is known that the linear system $Lx = b$ can be solved in time almost linear in the number of edges in Γ [138, 148]. Applied to (2.14), this means that we can compute $L^\dagger$ in $O((K+M)|E(\Gamma)|)$ time (up to logarithmic factors). Note that, even if L is dense, it is still possible to speed up the inversion (say, compared to a direct Gaussian elimination) using the formula [75, 117]:

$$L^\dagger = [L + (K+M)^{-1}\mathbf{1}\mathbf{1}^T]^{-1} - (K+M)^{-1}\mathbf{1}\mathbf{1}^T.$$

The speed up in this case is however in terms of the absolute run time. The overall complexity is still $O((K+M)^3)$, but with smaller constants. We note that it is also possible to speed up the inversion by exploiting the bipartite nature of Γ [75], although we have not used this in our implementation.

The complexity of the eigenvector computation is $O(M^3 d^3)$, while that of the orthogonal rounding is $O(Md^3)$. The total complexity of GRET-SPEC, say, using a linear-time Laplacian inversion, is (up to logarithmic factors)

$$O\left(|E(\Gamma)|(K+M) + (Md)^3\right).$$

The main computational blocks in GRET-SDP are identical to that in GRET-SPEC, plus the SDP computation. The SDP solution can be computed in polynomial time

using interior-point programming [163]. In particular, the complexity of computing an ε-accurate solution using interior-point solvers such as SDPT3 [141] is $O((Md)^{4.5}\log(1/\varepsilon))$. It is possible to lower this complexity by exploiting the particular structure of (P_2). For example, notice that the constraint matrices in (P_2) have at most one non-zero coefficient. Using the algorithm in [72], one can then bring down the complexity of the SDP to $O((Md)^{3.5}\log(1/\varepsilon))$. By considering a penalized version of the SDP, we can use first-order solvers such as TFOCS [12] to further cut down the dependence on M and d to $O((Md)^3\varepsilon^{-1})$, but at the cost of a stronger dependence on the accuracy. The quest for efficient SDP solvers is currently an active area of research. Fast SDP solvers have been proposed that exploit either the low-rank structure of the SDP solution [25, 82] or the simple form of the linearity constraints in (P_2) [156]. More recently, a sublinear time approximation algorithm for SDP was proposed in [54]. The complexity of GRET-SDP using a linear-time Laplacian inversion and an interior-point SDP solver is thus

$$O\left(|E(\Gamma)|(K+M)+(Md)^{4.5}\log(1/\varepsilon)+(Md)^3\right).$$

For problems where the size of the SDP variable is within 150, we can solve (P_2) in reasonable time on a standard PC using SDPT3 [141] or CVX [64]. We use CVX for the numerical experiments in Section 2.5 that involve small-to-moderate sized SDP variables. For larger SDP variables, one can use the low-rank structure of (P_2) to speed up the computation. In particular, we were able to solve for SDP variables of size up to 2000×2000 using SDPLR [25] that uses low-rank based heuristics.

2.2 Rigidity and exact recovery of GRET-SPEC and GRET-SDP

We now examine conditions on the membership graph under which the proposed spectral and convex relaxations can recover the global coordinates from the knowledge of the clean local coordinates (and the membership graph). More precisely, let $\bar{x}_1, \ldots, \bar{x}_K$ be the true coordinates of a point cloud in $\mathbb{R}^d$. Suppose that the point cloud is divided into patches whose membership graph is Γ, and that we are provided the measurements

$$x_k^{(i)} = \bar{O}_i^T(\bar{x}_k - \bar{t}_i) \qquad (k,i) \in E(\Gamma), \tag{2.25}$$

for some $\bar{O}_i \in \mathbb{O}(d)$ and $\bar{t}_i \in \mathbb{R}^d$. The patch-stress matrix C is constructed from Γ and the clean measurements (2.25). The question is under what conditions on Γ can $\bar{x}_1, \ldots, \bar{x}_K$ be recovered by our algorithm? We will refer to this as *exact recovery*.

In order to determine the conditions of exact recovery, we need to introduce some tools from rigidity theory. Rigidity theory studies the condition for a point-set to admit a unique configuration (up to a global transformation), when the point-set is subjected to pairwise geometric constraints, e.g. pairwise distances. To study exact recovery, here we introduce two rigidity notions: *affine rigidity* and *universal rigidity*.

2.2.1 Affine rigidity and universal rigidity

We now formally define the notion of affine rigidity. Although phrased differently, it is in fact identical to the definitions in [168, 59]. Henceforth, by affine transform, we will mean the group of non-singular affine maps on $\mathbb{R}^d$. Affine rigidity is a property of the patch-graph Γ and the local coordinates $(x_k^{(i)})$. In keeping with [59], we will together call these the *patch framework* and denote it by $\Theta = (\Gamma, (x_k^{(i)}))$.

Definition 2.2.1 (Affine Rigidity)**.** *Let $y_1, \ldots, y_K \in \mathbb{R}^d$ be such that, for some affine*

transforms $\rho_1, \ldots, \rho_M$,

$$y_k = \rho_i(x_k^{(i)}) \qquad (k,i) \in E(\Gamma).$$

The patch framework $\Theta = (\Gamma, (x_k^{(i)}))$ *is affinely rigid if* $y_1, \ldots, y_K$ *is identical to* $\bar{x}_1, \ldots \bar{x}_K$ *up to a global affine transform.*

Just as we defined affine rigidity earlier, we can phrase universal rigidity for a patch system as follows [61].

Definition 2.2.2 (Universal Rigidity)**.** *Let* $x_1, \ldots, x_K$ *be points in* $\mathbb{R}^s (s \geq d)$ *such that, for some orthogonal* $O_i \in \mathbb{R}^{s \times d}$ *and* $t_i \in \mathbb{R}^s$,

$$x_k = O_i x_k^{(i)} + t_i \qquad (k,i) \in E.$$

We say that the patch framework $\Theta = (\Gamma, (x_k^{(i)}))$ *is universally rigid in* $\mathbb{R}^d$ *if for any such* (x_k)*, we have* $x_k = \Omega \bar{x}_k$ *for some orthogonal* $\Omega \in \mathbb{R}^{s \times d}$.

By orthogonal Ω we mean that the columns of Ω are orthogonal and of unit norm. As we shall see later, the universal rigidity of a patch system is closely related to the concept of universal rigidity in distance geometry.

2.2.2 Exact recovery

With the definitions from rigidity theory we are now ready to give conditions on exact recovery. Define

$$\bar{Z} = \begin{bmatrix} \bar{x}_1 & \cdots & \bar{x}_K & \bar{t}_1 & \cdots & \bar{t}_M \end{bmatrix} \in \mathbb{R}^{d \times (K+M)},$$

and

$$\bar{O} = [\bar{O}_1 \ \cdots \ \bar{O}_M] \in \mathbb{R}^{d \times Md}.$$

Then, exact recovery means that for some $\Omega \in \mathbb{O}(d)$ and $t \in \mathbb{R}^d$,

$$Z^\star = \Omega \bar{Z} + t\mathbf{1}^T.$$

Henceforth, we will always assume that Γ is connected (clearly one cannot have exact recovery otherwise).

Before proceeding, we remark that the conditions for exact recovery have previously been examined by Zha and Zhang [168] in the context of tangent-space alignment in manifold learning, and later by Gortler et al. [59] using affine rigidity. Moreover, the authors in [59] relate this notion of rigidity to other standard notions of rigidity, and provide conditions on a certain hypergraph constructed from the patch system that can guarantee affine rigidity. In this section we relate these rigidity results to the properties of the membership graph Γ (and the patch-stress matrix C). We note that the authors in [59] directly examine the uniqueness of the global coordinates, while we are concerned with the uniqueness of the patch transforms obtained by solving (P_1) and (P_2). The uniqueness of the global coordinates is then immediate:

Proposition 2.2.3 (Uniqueness and Exact Recovery)**.** *If* (P_1) *and* (P_2) *have unique solutions, then GRET-SPEC and GRET-SDP return* $\bar{x}_1, \ldots, \bar{x}_K$ *up to a global rigid transform.*

At this point, we note that if a patch has less than $d+1$ points, then even when $\bar{x}_1, \ldots, \bar{x}_K$ are the unique set of coordinates that satisfy 2.25, we cannot guarantee $\bar{O}_1, \ldots, \bar{O}_M$ and $\bar{t}_1, \ldots, \bar{t}_M$ to be unique. Therefore, we will work under the mild assumption that each patch has at least $d+1$ non-degenerate points, so that the patch transforms are uniquely determined from the global coordinates.

Since each patch has $d+1$ points, we now give a characterization of affine rigidity that will be useful later on.

Proposition 2.2.4. *A patch framework* $\Theta = (\Gamma, (x_k^{(i)}))$ *is affinely rigid if and only if for any* $F \in R^{d \times Md}$ *such that* $\mathrm{Tr}(CF^TF) = 0$ *we must have* $F = A\bar{O}$ *for some non-singular* $A \in R^{d \times d}$.

Before proceeding to the proof, note that $\bar{O}$ and $\bar{G} = \bar{O}^T\bar{O}$ are solutions of (P_1) and (P_2) (this was the basis of Proposition 2.2.3), and the objective in either case is zero. Indeed, from (2.25), we can write $\bar{Z}L = \bar{O}B$. Since Γ is connected,

$$\bar{Z} = \bar{O}BL^{\dagger} + t\mathbf{1}^T \qquad (t \in \mathbb{R}^d). \tag{2.26}$$

Using (2.26), it is not difficult to verify that $\phi(\bar{Z}, \bar{O}) = \mathrm{Tr}(C\bar{G})$. Moreover, it follows from (2.25) that $\phi(\bar{Z}, \bar{O}) = 0$. Therefore,

$$\mathrm{Tr}(C\bar{G}) = \mathrm{Tr}(C\bar{O}^T\bar{O}) = 0. \tag{2.27}$$

Using an identical line of reasoning, we also record another fact. Let $F = [F_1, \ldots, F_M]$ where each $F_i \in \mathbb{R}^{d\times d}$. Suppose there exists $y_1, \ldots, y_K \in \mathbb{R}^d$ and $t_1, \ldots, t_M \in \mathbb{R}^d$ such that

$$y_k = F_i x_k^{(i)} + t_i \qquad (k, i) \in E(\Gamma). \tag{2.28}$$

Then $Y = [y_1, \ldots, y_K, t_1, \ldots, t_M] \in \mathbb{R}^{d\times(K+M)d}$ satisfies

$$Y = FBL^{\dagger} + t\mathbf{1}^T \tag{2.29}$$

and $\mathrm{Tr}(CF^TF) = 0$.

of Proposition 2.2.4. For any F such that $\mathrm{Tr}(CF^TF) = 0$, letting $[y_1, \ldots, y_K, t_1, \ldots, t_M] = FBL^{\dagger}$, we have 2.28. By the affine rigidity assumption, we must then have $y_k = A\bar{x}_k + t$ for some non-singular $A \in \mathbb{R}^{d\times d}$ and $t \in \mathbb{R}^d$. Since each patch contains $d+1$ non-degenerate points, it follows that $F = A\bar{O}$.

In the other direction, assume that $y_1, \ldots, y_K \in \mathbb{R}^d$ satisfy 2.28. Then we know $\mathrm{Tr}(CF^TF) = 0$ and hence $F = A\bar{O}$ for some non-singular A. Using 2.29, we immediately have $y_k = A\bar{x}_k$. □

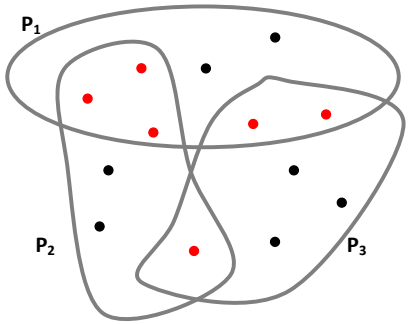


Figure 2.2: Instance of three overlapping patches, where the overlapping points are shown in red. In this case, P_3 cannot be registered with either P_1 or P_2 due to insufficient overlap. Therefore, the patches cannot be localized in two dimension using, for example, [169, 42] that work by registering pairs of patches. The patches can however be registered using GRET-SPEC and GRET-SDP since the ordered patches P_1, P_2, P_3 form a graph lateration in $\mathbb{R}^2$.

Note that $\mathrm{Tr}(CF^TF) = 0$ implies that the rows of F are in the null space of C. Therefore, the combined facts that $\mathrm{Tr}(CF^TF) = 0$ and $F = A\bar{O}$ for some non-singular $A \in R^{d\times d}$ is equivalent to saying that null space of C is within the row span of $\bar{O}$. The following result then follows as a consequence of 2.2.4.

Corollary 2.2.5. *A patch framework $\Theta = (\Gamma, (x_k^{(i)}))$ is affinely rigid if and only if the rank of C is $(M-1)d$.*

The corollary gives an easy way to check for affine rigidity. However, it is not clear what construction of Γ will ensure such property. In [168], the notion of graph *lateration* was introduced that guarantees affine rigidity: Γ is said to be a graph lateration (simply laterated) if there exists an reordering of the patch indices such that, for every $i \geq 2$, P_i and $P_1 \cup \cdots \cup P_{i-1}$ have at least $d+1$ non-degenerate nodes in common. An example of a graph lateration is shown in Figure 2.2.

Theorem 2.2.6 ([168])**.** *If Γ is laterated and the local coordinates are non-degenerate then the framework Θ is affinely rigid.*

Next, we turn to the exact recovery conditions for (P_2).The appropriate notion of

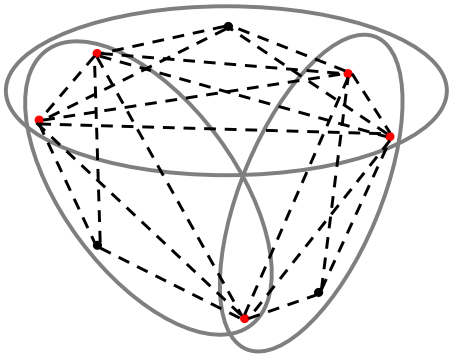


Figure 2.3: This shows the body graph for a 3-patch system. The edges of the body graph are obtained by connecting points that belong to the same patch. The edges within a given patch are marked with the same color. GRET-SDP can successfully register all the patches if the body graph is rigid in a certain sense.

rigidity in this case is that of universal rigidity. Following exactly the same arguments used to establish 2.2.4, one can derive the following.

Proposition 2.2.7. *A patch framework* $\Theta = (\Gamma, (x_k^{(i)}))$ *is universally rigid in* $\mathbb{R}^d$ *if and only if for any* $O \in \mathbb{R}^{s\times d}(s \geq d)$ *such that* $\mathrm{Tr}(CO^TO) = 0$ *we must have* $O = \Omega\bar{O}$ *for some orthogonal* $\Omega \in \mathbb{R}^{s\times d}$.

The question then is under what conditions is the patch framework universally rigid? This was also addressed in [59] using a graph construction derived from Γ called the *body graph*. This given by $\Gamma_B = (V_B, E_B)$, where $V_B = \{1, 2, \ldots, K\}$ and $(k, l) \in E_B$ if and only if x_k and x_l belong to the same patch (cf. Figure 2.3). Next, the following distances are associated with Γ_B:

$$d_{kl} = \|x_k^{(i)} - x_l^{(i)}\| \qquad (k, l) \in E_B, \tag{2.30}$$

where $x_k, x_l \in P_i$, say. Note that the above assignment is independent of the choice of patch. A set of points $(x_k)_{k\in V}$ in $\mathbb{R}^l$ is said to be a *realization* of $\{d_{kl} : (k, l) \in E\}$ in $\mathbb{R}^l$ if $d_{kl} = ||x_k - x_l||$ for $(k, l) \in E$.

It is shown in [59] that $\Theta = (\Gamma, (x_k^{(i)}))$ is universally rigid if and only if Γ_B with

distances $\{d_{kl} : (k,l) \in E\}$ has a unique realization in $\mathbb{R}^s$ for all $s \geq d$. Moreover, in such situation, using the distances as the constraints, an SDP relaxation was proposed in [137] for finding the unique realization. We note that although the SDP in [137] has the same condition for exact recovery as (P_2), it is computationally more demanding than (P_2) since the number of variables is of $O(K^2)$ for this SDP, instead of $O(M^2)$ as in (P_2) (for most applications, $M << K$). Moreover, as we will see shortly, (P_2) also enjoys some stability properties which has not been observed for the SDP in [137].

Finally, we note that universal rigidity is a weaker condition on Γ than affine rigidity.

Theorem 2.2.8 ([137], Theorem 2)**.** *If a patch framework is affinely rigid, then it is universally rigid.*

In [59], it was also shown that the reverse implication is not true using an counter-example for which the patch framework fails to be affinely rigid, but for which the body graph (a Cauchy polygon) has an unique realization in any dimension [36]. This means that GRET-SDP can solve a bigger class of problems than GRET-SPEC, which is perhaps not surprising.

2.3 Randomized rank test for affine rigidity

Corollary 2.2.5 tells us by checking the rank of the patch stress matrix C, we can tell whether a patch framework is affinely rigid. In this regard, the patch-stress matrix serves the same purpose as the so-called alignment matrix in [168] and the affinity matrix in [59]. The only difference is that the kernel of C represents the degree of freedom of the affine transform, whereas kernel of alignment or affinity matrix directly tell us the degree of freedom of the point coordinates. As suggested in [59], an efficient randomized test for affine rigidity using the concept of affinity matrix can be easily derived. In this section, we describe a randomized test based on patch stress matrix, which parallels the proposal in [59]. This procedure is also similar in spirit to the randomized tests for

generic local rigidity by Hendrickson [73], for generic global rigidity by Gortler et al. [60], and for matrix completion by Singer and Cucuringu [135].

Let us continue to denote the patch-stress matrix obtained from Γ and the measurements (2.25) by C. We will use C_0 to denote the patch-stress matrix obtained from the same graph Γ, but using the (unknown) original coordinates as measurements, namely,

$$x_k^{(i)} = \bar{x}_k \qquad (k,i) \in \Gamma. \tag{2.31}$$

The advantage of working with C_0 over C is that the former can be computed using just the global coordinates, while the latter requires the knowledge of the global coordinates as well as the clean transforms. In particular, this only requires us to simulate the global coordinates. Since the coordinates of points in a given patch are determined up to a rigid transform, we claim the following (cf. Section 2.7.1 for a proof).

Proposition 2.3.1 (Rank equivalence)**.** *For a fixed Γ, C and C_0 have the same rank.*

In other words, the rank of C_0 can be used to certify exact recovery. The proposed test is based on Proposition 2.7.1, and the fact that if two different *generic* configurations are used as input in (2.31) (for the same Γ), then the patch-stress matrices they produce would have the same rank. By generic, we mean that the coordinates of the configuration do not satisfy any non-trivial algebraic equation with rational coefficients [60].

The complete test is described in Algorithm 3. "Randomized Rank Test" (`RRT`). Note that the main computations in `RRT` are the Laplacian inversion (which is also required for the registration algorithm) and the rank computation.

2.4 Stability Analysis

We have so far studied the problem of exact recovery from noiseless measurements. In practice, however, the measurements are invariably noisy. This brings us to the question

Algorithm 3 RRT

Require: Membership graph Γ, and dimension d.
Ensure: Exact recovery certificate for GRET-SDP.
1: Build L using Γ, and compute $L^{\dagger}$.
2: Randomly pick $\{x_1, \ldots, x_K\}$ from the unit cube in $\mathbb{R}^d$, where $K = |V_x(\Gamma)|$.
3: $x_k^{(i)} \leftarrow x_k$ for every $(k,i) \in E(\Gamma)$.
4: $C_0 \leftarrow D - BL^{\dagger}B^T$.
5: **if** $\operatorname{rank}(C_0) = (M-1)d$ **then**
6: Positive certificate for GRET-SPEC and GRET-SDP.
7: **else**
8: Negative certificate for GRET-SPEC.
9: GRET-SDP cannot be certified.
10: **end if**

of stability, namely how stable are GRET-SPEC and GRET-SDP to perturbations in the measurements? Numerical results (to be presented in the next Section) show that both the relaxations are indeed quite stable to perturbations. In particular, the reconstruction error degrades quite gracefully with the increase in noise (reconstruction error is the gap between the outputs with clean and noisy measurements). In this Section, we try to quantify these empirical observations. In particular, we show that, for a specific noise model, the reconstruction error grows at most linearly with the level of noise.

The noise model we consider is the "bounded" noise model. Namely, we assume that the measurements are obtained through bounded perturbations of the clean measurements in (2.25). More precisely, we suppose that we have a membership graph Γ, and that the observed local coordinates are of the form

$$x_k^{(i)} = \bar{O}_i^T(\bar{x}_k - \bar{t}_i) + \epsilon_k^{(i)}, \qquad \|\epsilon_k^{(i)}\| \leq \varepsilon \quad (k,i) \in E(\Gamma). \tag{2.32}$$

In other words, every coordinate measurement is offset within a ball of radius ε around the clean measurements. Here, ε is a measure of the noise level per measurement. In particular, $\varepsilon = 0$ corresponds to the case where we have the clean measurements (2.25).

Since the coordinates of points in a given patch are determined up to a rigid transform,

it is clear that the above problem is equivalent to the one where the measurements are

$$x_k^{(i)} = \bar{x}_k + \epsilon_k^{(i)}, \qquad \|\epsilon_k^{(i)}\| \leq \varepsilon \quad (k,i) \in E(\Gamma). \tag{2.33}$$

By equivalent, we mean that the reconstruction errors obtained using either (2.32) or (2.33) are equal. The reason we use the latter measurements is that the analysis in this case is much more simple.

The reconstruction error is defined as follows. Generally, let $Z^\star$ be the output of Algorithms 1 and 2 using (2.33) as input, and let

$$Z_0 \stackrel{\text{def}}{=} [\bar{x}_1 \cdots \bar{x}_K \; 0 \cdots 0] \in \mathbb{R}^{d \times (K+M)}, \tag{2.34}$$

where we assume that the centroid of $\{\bar{x}_1, \cdots, \bar{x}_K\}$ is at the origin.

Ideally, we would require that $Z^\star = Z_0$ (up to a rigid transformation) when there is no noise, that is, when $\varepsilon = 0$. This is the exact recovery phenomena that we considered earlier. In general, the gap between Z_0 and $Z^\star$ is a measure of the reconstruction quality. Therefore, we define the reconstruction error to be

$$\eta = \min_{\Theta \in \mathbb{O}(d)} \|Z^\star - \Theta Z_0\|_F.$$

Note that we are not required to factor out the translation since Z_0 is centered by construction.

Our main results are the following.

Theorem 2.4.1 (Stability of GRET-SPEC)**.** *Assume that R is the radius of the smallest Euclidean ball that encloses the clean configuration $\{\bar{x}_1, \ldots, \bar{x}_K\}$. For fixed noise level $\varepsilon \geq 0$ and membership graph Γ, suppose we input the noisy measurements* (2.33) *to GRET-SPEC.*

If $\text{rank}(C_0) = (M-1)d$*, then we have the following bound for GRET-SPEC*:

$$\eta \leq \frac{|E(\Gamma)|^{1/2}}{\lambda_2(L)}(K_1\varepsilon + K_2\varepsilon^2),$$

where

$$K_1 = \frac{8\pi R}{\mu_{d+1}(C)}\sqrt{2MK|E(\Gamma)|(2+K)d(d+1)}\left(4R\frac{\sqrt{K|E(\Gamma)|}}{\lambda_2(L)}+1\right)+\sqrt{2+K+M}.$$

and

$$K_2 = \frac{8\pi R}{\mu_{d+1}(C)}\sqrt{2MK|E(\Gamma)|(2+K)d(d+1)}\left(2\frac{\sqrt{K|E(\Gamma)|}}{\lambda_2(L)}+1\right).$$

Here $\lambda_2(L)$ *is the second smallest eigenvalue of* L.

We assume here that $\mu_{d+1}(C)$ is non-zero[2]. The bounds here are in fact quite loose. Note that when $\varepsilon = 0$, then by the admissibility assumption $\mu_{d+1}(C) > 0$, and we recover the perfect reconstruction results for GRET-SPEC.

Theorem 2.4.2 (Stability of GRET-SDP)**.** *Under the conditions of Theorem 2.4.1, we have the following for GRET-SDP*:

$$\eta \leq \frac{|E(\Gamma)|^{1/2}}{\lambda_2(L)}\left[32\sqrt{2d(d+1)(2+K)|E(\Gamma)|}\mu_{d+1}^{-1/2}(C_0)R+\sqrt{2+K+M}\right]\varepsilon.$$

The bounds are again quite loose. The main point is that the reconstruction error for GRET-SDP is within a constant factor of the noise level. In particular, when $\varepsilon = 0$ (measurements are clean), we recover the perfect reconstruction results.

The rest of this Section is devoted to the proofs of Theorem 2.4.1 and 2.4.2. First, we introduce some notations.

Notations. Note that the patch-stress matrix in (P_1) is computed from the noisy

[2]Numerical experiments suggest that this is indeed the case if $\text{rank}(C_0) = (M-1)d$. In fact, we notice a growth in the eigenvalue with the increase in noise level. We have however not been able to prove this fact.

measurements (2.33), and the same patch-stress matrix is used in (P_2). The quantities $G^\star, W^\star, O^\star$, and $Z^\star$ are as defined in Algorithms 1 and 2 . We continue to denote the clean patch-stress matrix by C_0. Define

$$O_0 \stackrel{\text{def}}{=} [I_d \cdots I_d] \quad \text{and} \quad G_0 \stackrel{\text{def}}{=} O_0^T O_0.$$

Let $e_1, \ldots, e_d$ be the standard basis vectors of $\mathbb{R}^d$, and let e be the all-ones vector of length M. Define

$$s_i \stackrel{\text{def}}{=} \frac{1}{\sqrt{M}} e \otimes e_i \in \mathbb{R}^{Md} \qquad (1 \le i \le d). \tag{2.35}$$

Note that every $d \times d$ block of G_0 is I_d, and that we can write

$$G_0 = \sum_{i=1}^{d} M s_i s_i^T. \tag{2.36}$$

We first present an estimate that applies generally to both algorithms. The proof is provided in Section 2.7.2.

Proposition 2.4.3 (Basic estimate)**.** *Let R be the radius of the smallest Euclidean ball that encloses the clean configuration. Then, for any arbitrary* Θ*,*

$$\|Z^\star - \Theta Z_0\|_F \le \frac{|E(\Gamma)|^{1/2}}{\lambda_2(L)} \Big[R(2+K)^{1/2} \|O^\star - \Theta O_0\|_F + \varepsilon (2+K+M)^{1/2} \Big]. \tag{2.37}$$

In other words, the reconstruction error in either case is controlled by the rounding error:

$$\delta = \min_{\Theta \in \mathbb{O}(d)} \|O^\star - \Theta O_0\|_F. \tag{2.38}$$

The rest of this Section is devoted to obtaining a bound on δ for GRET-SPEC and GRET-SDP. In particular, we will show that δ is of the order of ε in either case. Note that the key difference between the two algorithms arises from the eigenvector rounding, namely the assignment of the "unrounded" orthogonal transform $W^\star$ (respectively from the

patch-stress matrix and the optimal Gram matrix). The analysis in going from $W^\star$ to the rounded orthogonal transform $O^\star$, and subsequently to $Z^\star$, is however common to both algorithms.

We now bound the error in (2.38) for both algorithms. Note that we can generally write

$$W^\star = \left[\sqrt{\alpha_1}u_1 \cdots \sqrt{\alpha_d}u_d\right]^T,$$

where $u_1, \ldots, u_d$ are orthonormal. In GRET-SPEC, each $\alpha_i = M$, while in GRET-SDP we set α_i using the eigenvalues of $G^\star$.

Our first result gives a control on the quantities obtained using eigenvector rounding in terms of their Gram matrices.

Lemma 2.4.4 (Eigenvector rounding)**.** *There exist $\Theta \in \mathbb{O}(d)$ such that*

$$\|W^\star - \Theta O_0\|_F \leq \frac{4}{\sqrt{M}} \|W^{\star T} W^\star - G_0\|_F.$$

Next, we use a result by Li [98] to get a bound on the error after orthogonal rounding.

Lemma 2.4.5 (Orthogonal rounding)**.** *For arbitrary $\Theta \in \mathbb{O}(d)$,*

$$\|O^\star - \Theta O_0\|_F \leq 2\sqrt{d+1}\, \|W^\star - \Theta O_0\|_F.$$

The proofs of Lemma 2.4.4 and 2.4.5 are provided in Appendices 2.7.3 and 2.7.4. At this point, we record a result from [107] which is repeatedly used in the proof of these lemmas and elsewhere.

Lemma 2.4.6 (Mirsky, [107])**.** *Let $|||\cdot|||$ be some unitarily invariant norm, and let $A, B \in \mathbb{R}^{n\times n}$. Then*

$$|||\operatorname{diag}(\sigma_1(A) - \sigma_1(B), \cdots, \sigma_n(A) - \sigma_n(B))||| \leq |||A - B|||.$$

In particular, the above result holds for the Frobenius and spectral norms.

By combining Lemma 2.4.4 and 2.4.5, we have the following bound for (2.38):

$$\delta \leq 8\sqrt{\frac{d+1}{M}}\,\|W^{\star T}W^{\star} - G_0\|_F. \tag{2.39}$$

We now bound the quantity on the right in (2.39) for GRET-SPEC and GRET-SDP.

2.4.1 Bound for GRET-SPEC

For the spectral relaxation, this can be done using the Davis-Kahan theorem [16]. Note that from (2.18), we can write

$$\frac{1}{M}(W^{\star T}W^{\star} - G_0) = \sum_{i=1}^{d} r_i r_i^T - \sum_{j=1}^{d} s_j s_j^T. \tag{2.40}$$

Following [16, Ch. 7], let A be some symmetric matrix and S be some subset of the real line. Denote $P_A(S)$ to be the orthogonal projection onto the subspace spanned by the eigenvectors of A whose eigenvalues are in S. A particular implication of the Davis-Kahan theorem is that

$$\|\, P_A(S_1) - P_B(S_2)\,\|_{\text{sp}} \leq \frac{\pi}{2\rho(S_1^c, S_2)}\|A-B\|_{\text{sp}}, \tag{2.41}$$

where S_1^c is the complement of S_1, and $\rho(S_1,S_2) = \min\{|u-v| : u \in S_1, v \in S_2\}$.

In order to apply (2.41) to (2.40), set $A = C, B = C_0, S_1 = [\mu_1(C),\ \mu_d(C)]$, and $S_2 = \{0\}$. If $\text{rank}(C_0) = (M-1)d$, then $P_B(S_2) = \sum_{j=1}^{d} s_j s_j^T$. Applying (2.41), we get

$$\|W^{\star T}W^{\star} - G_0\|_{\text{sp}} \leq \frac{M\pi}{2\mu_{d+1}(C)}\|C - C_0\|_F. \tag{2.42}$$

Now, it is not difficult to verify that for the noise model (2.33),

$$\|C - C_0\|_F \leq 2\sqrt{K|E(\Gamma)|}\left[\left(4R\frac{\sqrt{K|E(\Gamma)|}}{\lambda_2(L)} + 1\right)\varepsilon + \left(2\frac{\sqrt{K|E(\Gamma)|}}{\lambda_2(L)} + 1\right)\varepsilon^2\right]. \tag{2.43}$$

Combining Proposition 2.4.3 with (2.39),(2.42), and (2.43), we arrive at Theorem 2.4.1.

2.4.2 Bound for GRET-SDP

To analyze the bound for GRET-SDP, we require further notations. Recall (2.35), and let S be the space spanned by $\{s_1, \ldots, s_d\} \subset \mathbb{R}^{Md}$, and let $\bar{S}$ be the orthogonal complement of S in $\mathbb{R}^{Md}$. In the sequel, we will be required to use matrix spaces arising from tensor products of vector spaces. In particular, given two subspaces U and V of $\mathbb{R}^{Md}$, denote by $U \otimes V$ the space spanned by the rank-one matrices $\{uv^T : u \in U, v \in V\}$. In particular, note that G_0 is in $S \otimes S$.

Let $A \in \mathbb{R}^{Md \times Md}$ be some arbitrary matrix. We can decompose it into

$$A = P + Q + T \tag{2.44}$$

where

$$P \in S \otimes S,\ Q \in (S \otimes \overline{S}) \cup (\overline{S} \otimes S),\ \text{and}\ T \in \overline{S} \otimes \overline{S}.$$

We record a result about this decomposition from Wang and Singer [152].

Lemma 2.4.7 ([152], pg. 7)**.** *Suppose* $G_0 + \Delta \succeq 0$ *and* $\Delta_{ii} = 0$ $(1 \leq i \leq M)$*. Let* $\Delta = P + Q + T$ *as in* (2.44)*. Then*

$$T \succeq 0, \quad \textit{and} \quad P_{ij} = -\frac{1}{M} \sum_{l=1}^{M} T_{ll} \quad (1 \leq i, j \leq M).$$

It is not difficult to verify that $\mathrm{Tr}(C_0 G_0) = 0$ and that $C_0 \succeq 0$. From (2.36), we have

$$0 = \mathrm{Tr}(C_0 G_0) = \sum_{i=1}^{d} s_i^T C_0 s_i \geq 0.$$

Since each term in the above sum is non-negative, $C_0 s_i = 0$ for $1 \leq i \leq d$. In other words, S is contained in the null space of C_0. Moreover, if $\mathrm{rank}(C_0) = (M-1)d$, then

S is exactly the null space of C_0. Based on this observation, we give a bound on the residual T.

Proposition 2.4.8 (Bound on the residual)**.** *Suppose that* $\mathrm{rank}(C_0) = (M-1)d$*. Decompose* $\Delta = P + Q + T$ *as in* (2.44)*. Then*

$$\mathrm{Tr}(T) \leq 4\mu_{d+1}^{-1}(C_0)|E(\Gamma)|\varepsilon^2. \tag{2.45}$$

Proof. The main idea here is to compare the objective in (P_0) with the trace of T. To do so, we introduce the following notations. Let $\lambda_1, \cdots, \lambda_{Md}$ be the full set of eigenvalues of $G^\star$ sorted in non-increasing order, and $q_1, \ldots, q_{Md}$ be the corresponding eigenvectors. Define

$$O^{\star\star} \stackrel{\text{def}}{=} \left[\sqrt{\lambda_1}q_1 \ \cdots \ \sqrt{\lambda_{Md}}q_{Md}\right]^T \in \mathbb{R}^{Md\times Md},$$

and $O_i^{\star\star}$ to be the i-th $Md \times d$ block of $O^{\star\star}$, that is, $O^{\star\star} \stackrel{\text{def}}{=} [O_1^{\star\star} \ \cdots \ O_M^{\star\star}]$.

By construction, $G^\star = O^{\star\star T} O^{\star\star}$. Moreover, by feasibility,

$$G_{ii}^\star = O_i^{\star\star T} O_i^{\star\star} = I_d \quad (1 \leq i \leq M).$$

Thus the d columns of $O_i^{\star\star}$ form an orthonormal system in $\mathbb{R}^{Md}$. Now define

$$Z^{\star\star} \stackrel{\text{def}}{=} O^{\star\star} B L^\dagger \in \mathbb{R}^{Md\times(K+M)}.$$

In particular, we will use the fact that $(Z^{\star\star}, O^{\star\star})$ are the minimizers of the unconstrained program

$$\min_{(Z,O)} \sum_{(k,i)\in E(\Gamma)} \|Ze_{ki} - O_i x_k^{(i)}\|^2 \quad \text{s.t.} \quad Z \in \mathbb{R}^{Md\times(K+M)},\ O \in \mathbb{R}^{Md\times Md}. \tag{2.46}$$

Note that $\mathrm{Tr}(C_0 G^\star) = \mathrm{Tr}(C_0(G_0 + \Delta)) = \mathrm{Tr}(C_0 T)$. Now, by Lemma (2.4.7), $T \succeq 0$.

Therefore, writing

$$T = \sum_i v_i v_i^T \quad (v_i \in \bar{S}),$$

we get

$$\mathrm{Tr}(C_0 T) = \sum_i v_i^T C_0 v_i \geq \mu_{d+1}(C_0) \sum_i v_i^T v_i = \mu_{d+1}(C_0)\mathrm{Tr}(T).$$

Therefore,

$$\mathrm{Tr}(T) \leq \mu_{d+1}^{-1}(C_0)\ \mathrm{Tr}(C_0 G^\star). \tag{2.47}$$

We are done if we can bound the term on the right. To do so, we note from (2.46) that

$$\mathrm{Tr}(C_0 G^\star) = \mathrm{Tr}(C_0 O^{\star\star T} O^{\star\star}) = \min_{Z \in \mathbb{R}^{Md \times K+M}} \sum_{(k,i)\in E(\Gamma)} \|Z e_{ki} - O_i^{\star\star} \bar{x}_k\|^2.$$

Therefore,

$$\mathrm{Tr}(C_0 G^\star) \leq \sum_{(k,i)\in E(\Gamma)} \|Z^{\star\star} e_{ki} - O_i^{\star\star} \bar{x}_k\|^2.$$

To bring in the error term, we write

$$Z^{\star\star} e_{ki} - O_i^{\star\star} \bar{x}_k = Z^{\star\star} e_{ki} - O_i^{\star\star} x_k^{(i)} + O_i^{\star\star} \epsilon_k^{(i)},$$

and use $\|x+y\|^2 \leq 2(\|x\|^2 + \|y\|^2)$ to get

$$\mathrm{Tr}(C_0 G^\star) \leq 2 \sum_{(k,i)\in E} \|Z^{\star\star} e_{ki} - O_i^{\star\star} x_k^{(i)}\|^2 + 2|E(\Gamma)|\varepsilon^2. \tag{2.48}$$

Finally, using the optimality of $(Z^{\star\star}, O^{\star\star})$ for (2.46), we have

$$\sum_{(k,i)\in E(\Gamma)} \|Z^{\star\star} e_{ki} - O_i^{\star\star} x_k^{(i)}\|^2 \leq \sum_{(k,i)\in E(\Gamma)} \|Z_0 e_{ki} - I_d x_k^{(i)}\|^2 \leq |E(\Gamma)|\varepsilon^2. \tag{2.49}$$

The desired result follows from (2.47), (2.48), and (2.49). □

Finally, we note that $\mathrm{Tr}(T)$ can be used to bound the difference between the Gram

matrices.

Proposition 2.4.9 (Trace bound). $\|W^{\star T}W^\star - G_0\|_F \le 2\sqrt{2Md\,\mathrm{Tr}(T)}$.

Proof. We will heavily use decomposition (2.44) and its properties. Let $G^\star = G_0 + \Delta$. By triangle inequality,

$$\begin{aligned}\|W^{\star T}W^\star - G_0\|_F &\le \|\sum_{i=d+1}^{Md} \lambda_i(G^\star)\, u_i u_i^T\|_F + \|\Delta\|_F \\ &= \|\,\mathrm{diag}(\lambda_{d+1}(G^\star),\ldots,\lambda_{Md}(G^\star))\,\|_F + \|\Delta\|_F.\end{aligned}$$

Moreover, since the bottom eigenvalues of G_0 are zero, it follows from Lemma 2.4.6 that the norm of the diagonal matrix is bounded by $\|\Delta\|_F$. Therefore,

$$\|W^{\star T}W^\star - G_0\|_F \le 2\|\Delta\|_F. \tag{2.50}$$

Fix $\{s_{d+1},\ldots,s_{Md}\}$ to be some orthonormal basis of $\bar{S}$. For arbitrary $A \in \mathbb{R}^{Md}$, let

$$A(p,q) = s_p^T A s_q \qquad (1 \le p,q \le Md).$$

That is, $(A(p,q))$ are the coordinates of A in the basis $\{s_1,\ldots,s_d\} \cup \{s_{d+1},\ldots,s_{Md}\}$.

Decompose $\Delta = P + Q + T$ as in (2.44). Note that P, Q, and T are represented in the above basis as follows: P is supported on the upper $d \times d$ diagonal block, T is supported on the lower $(M-1)d \times (M-1)d$ diagonal block, and Q on the off-diagonal blocks. The matrix G_0 is diagonal in this representation.

We can bound $\|P\|_F$ using Lemma 2.4.7,

$$\|P\|_F^2 = M^2\|P_{11}\|_F^2 = \|\sum_{l=1}^{M} T_{ll}\|_F^2 \le \Big[\mathrm{Tr}\Big(\sum_{l=1}^{M} T_{ll}\Big)\Big]^2 = \mathrm{Tr}(T)^2, \tag{2.51}$$

where we have used the properties $T \succeq 0$ and $T_{ll} \succeq 0$ $(1 \leq l \leq M)$. In particular,

$$\|T\|_F \leq \mathrm{Tr}(T). \tag{2.52}$$

On the other hand, since $G_0 + \Delta \succeq 0$, we have $(G_0 + \Delta)(p,q)^2 \leq (G_0 + \Delta)(p,p)(G_0 + \Delta)(q,q)$. Therefore,

$$\|Q\|_F^2 = 2\sum_{p=1}^{d}\sum_{q=d+1}^{Md} Q(p,q)^2 \leq 2\sum_{p=1}^{d}(G_0 + \Delta)(p,p)\sum_{q=d+1}^{Md} T(q,q).$$

Notice that $0 = \mathrm{Tr}(\Delta) = \mathrm{Tr}(T) + \mathrm{Tr}(P)$. Therefore,

$$\|Q\|_F^2 \leq 2Md\,\mathrm{Tr}(T) - 2\mathrm{Tr}(T)^2. \tag{2.53}$$

Combining (2.50), (2.51), (2.53), and (2.52), we get the desired bound. □

Putting together (2.39) with Propositions (2.4.3),(2.4.8), and (2.4.9), we arrive at Theorem (2.4.2).

2.5 Simulations

We now present some numerical results on multipatch registration using GRET-SPEC and GRET-SDP. In particular, we study the exact recovery and stability properties of the algorithm. We define the reconstruction error in terms of the root-mean-square deviation (RMSD) given by

$$\mathrm{RMSD} = \min_{\Omega \in \mathbb{O}(d), t \in \mathbb{R}^d} \left[\frac{1}{K}\sum_{k=1}^{K} \|Z_k^\star - \Omega \bar{x}_k - t\|^2\right]^{1/2}. \tag{2.54}$$

In other words, the RMSD is calculated after registering (aligning) the original and the reconstructed configurations. We use the SVD-based algorithm [8] for this purpose.

We first consider a few examples concerning the registration of three patches in $\mathbb{R}^2$, where we vary Γ by controlling the number of points in the intersection of the patches. We work with the clean data in (2.25) and demonstrate exact recovery (up to numerical precision) for different Γ.

In the left plot in Figure 2.4, we consider a patch system with $K = 10$ points. The points that belong to two or more patches are marked red, while the rest are marked black. The patches taken in the order P_1, P_2, P_3 form a lateration in this case. As predicted by Corollary 2.2.5 and Theorem 2.2.6, the rank of the patch-stress matrix C_0 for this system must be $2(3-1) = 4$. This is indeed confirmed by our experiment. We expect GRET-SPEC and GRET-SDP to recover the exact configuration. Indeed, we get a very small RMSD of the order of 1e-7 in this case. As shown in the figure, the reconstructed coordinates obtained using GRET-SDP perfectly match the original ones after alignment.

We next consider the example shown in the center plot in Figure 2.4. The patch system is not laterated in this case, but the rank of C_0 is 4. Again we obtain a very small RMSD of the order 1e-7 for this example. This example demonstrates that lateration is not necessary for exact recovery.

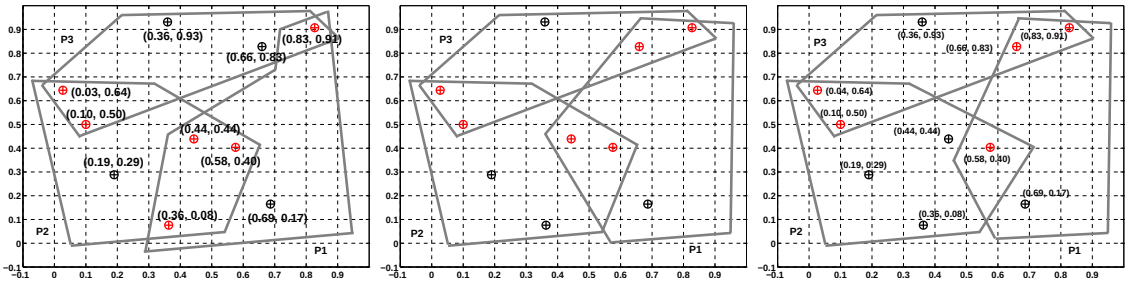


Figure 2.4: Instances of a three-patch systems in $\mathbb{R}^2$. Left: Patch system is laterated. Center: Patch system is not laterated but for which C_0 has rank 4. Right: The body graph is universally rigid but $\text{rank}(C_0) = 3$. The original coordinates are marked with $\circ$, and the coordinates reconstructed by GRET-SDP with $+$.

In the next example, we show that the condition $\text{rank}(C_0) = (M-1)d$ is not necessary for exact recovery using GRET-SDP. To do so, we use the fact that the universal rigidity of the body graph is both necessary and sufficient for exact recovery. Consider the example

shown in the right plot in Figure 2.4. This has barely enough points in the patch intersections to make the body graph universally rigid. Experiments confirm that we have exact recovery in this case. However, it can be shown that $\text{rank}(C_0) < (M-1)d = 4$.

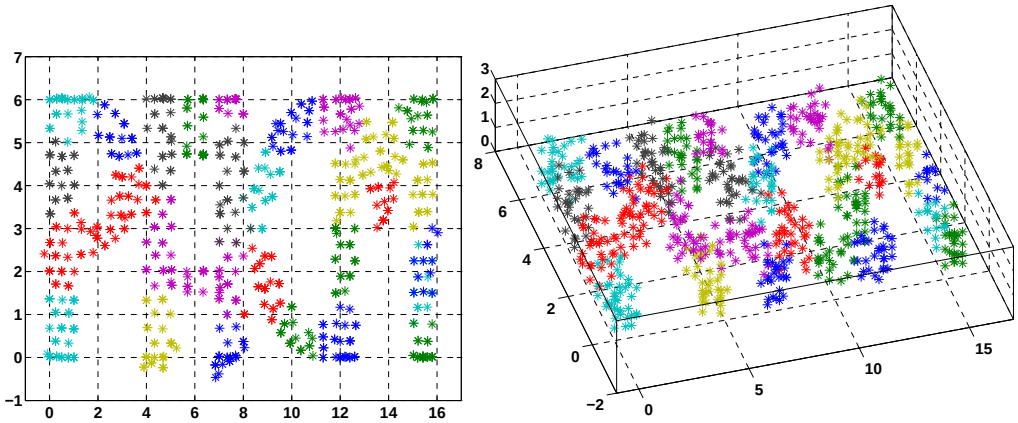


Figure 2.5: Disjoint clusters for the PACM point cloud. Each cluster is marked with a different color. The clusters are augmented to form overlapping patches which are then registered using GRET-SDP.

We now consider the structured PACM data in $\mathbb{R}^3$ shown in Figure 2.5. The are a total of 799 points in this example that are obtained by sampling the 3-dimensional PACM logo [42, 51]. To begin with, we divide the point cloud into $M = 30$ disjoint pieces (clusters) as shown in the figure. We augment each cluster into a patch by adding points from neighboring clusters. We ensure that there are sufficient common points in the patch system so that C_0 has rank $(M-1)d = 87$. We generate the measurements using the bounded noise model in (2.33). In particular, we perturb the clean coordinates using uniform noise over the hypercube $[-\varepsilon, \varepsilon]^d$. For the noiseless setting, the RMSD's obtained using GRET-SPEC and GRET-SDP are 3.3e-11 and 1e-6. The respective RMSD's when $\varepsilon = 0.5$ are 1.4743 and 0.3823. The results are shown in Figure 2.6.

In the final experiment, we demonstrate the stability of GRET-SDP and GRET-SPEC by plotting the RMSD against the noise level for the PACM data. We use the noise model in (2.33) and vary ε from 0 to 2 in steps of 0.1. For a fixed noise level, we average the RMSD over 20 noise realizations. The results are reported in the bottom plot in Figure 2.7. We see that the RMSD increases gracefully with the noise level. The result also shows

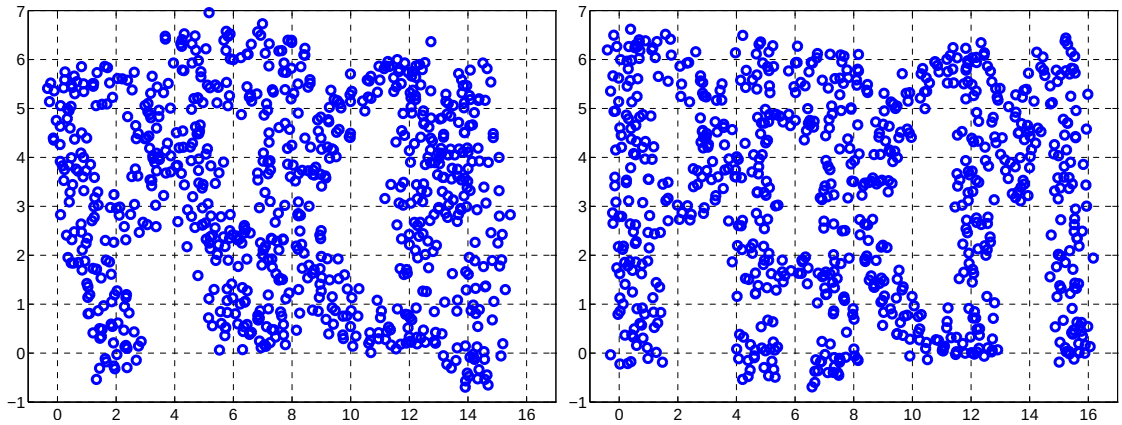


Figure 2.6: Reconstruction of the PACM data from corrupted patch coordinates ($\varepsilon = 0.5$). **Left**: GRET-SPEC, RMSD $= 1.4743$. **Right**: GRET-SDP, RMSD $= 0.3823$. The measurements were generated using the noise model in (2.33).

that the semidefinite relaxation is more stable than spectral relaxation, particularly at large noise levels. Also shown in the figure are the RMSD obtained using `GRET-MANOPT` with the solutions of GRET-SPEC and GRET-SDP as initialization. In particular, we used the trust region method provided in the `Manopt` toolbox [21] for solving the manifold optimization (P_0). For either initialization, we notice some improvement from the plots. It is clear that the manifold method relies heavily on the initialization, which is not surprising.

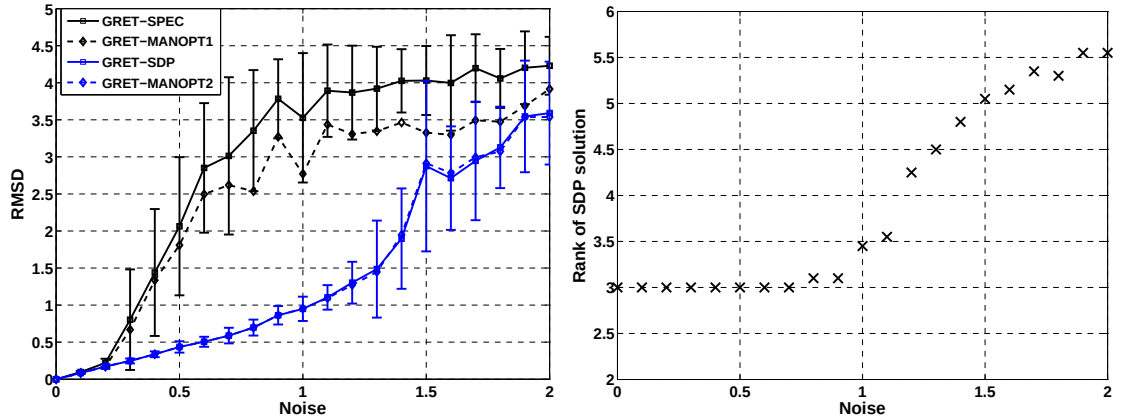


Figure 2.7: **Left**: RMSD versus noise level ε. `GRET-MANOPT1` (resp. `GRET-MANOPT2`) is the result obtained by refining the output of GRET-SPEC (resp. GRET-SDP) using manifold optimization. **Right**: Rank of $G^\star$ in GRET-SDP.

Finally, we plot the rank of the SDP solution $G^\star$ and notice an interesting phenomenon.

Up to a certain noise level, $G^\star$ has the desired rank and rounding is not required. This means that the relaxation gap is zero for the semidefinite relaxation, and that we can solve the original non-convex problem using GRET-SDP up to a certain noise threshold. It is therefore not surprising that the RMSD shows no improvement after we refine the SDP solution using manifold optimization. We have noticed that the rank of the SDP solution is stable with respect to noise for other numerical experiments as well (not reported here).

2.6 Distributed structural determination via GRET-SDP

We can readily use GRET to determine the protein structure under a divide-and-conquer procedure. First we construct an adjacency matrix $A \in \mathbb{R}^{K\times K}$ where $A(i,j) = 1$ indicates atoms i and j are related via some distance measurements that arise from bond lengths, bond angles, torsion angle restraints and NOE restraints. Our proposed algorithm consists of the following steps: (1) Partition the distance graph with the adjacency matrix A recursively using spectral clustering, (2) Embed using SNLSDP, (3) Stitch using GRET. When breaking the distance graph into patches, we want to minimize the number of inter-patch edges and maximize the edges within each patch, in order to retain as many distance measurements as possible within patches in the embedding phase. This can be achieve by the normalized cut (Ncut) algorithm proposed in [131] which splits a graph into two disjoint partitions of similar size while minimizing the inter-partition edges. To obtain multiple patches, we split the distance graph with adjacency matrix A recursively, i.e. after every splitting, we apply Ncut again to each of the resulting partitions. We recursively split the distance graph into disjoint partitions where each partition has about 50 atoms. At this point every partitions do not share common atoms with any other partition. Therefore for each partition, we add to it the atoms that are connected to the partition with sufficient distance measurements. We start by adding

those atoms that is most densely connected to the patch until the size of the patch is about 80. We pay little effort in ensuring whether there is sufficient distance restraint such that each patch will have a unique embedding, as in [41]. This is because to a certain degree the molecule is flexible, rendering obtaining rigid patches unrealistic. The hope is that the flexibility of the protein is not so large such that after SNLSDP embedding, the coordinates of each patch are still determined approximately up to a rigid transformation.

For simulation purpose, we take the coordinates of the structures deposited on the PDB as ground truth and denote them as $\bar{x}_1, \ldots, \bar{x}_K$. Then we randomly select hydrogen-hydrogen distances within 6 Å and simulate distance bounds by the equations

$$d_{ij}^{\text{low}} = \max(\|\bar{x}_i - \bar{x}_j\|(1-\epsilon_{ij}^{\text{low}}), 1.8) \qquad d_{ij}^{\text{up}} = \min(\|\bar{x}_i - \bar{x}_j\|(1+\epsilon_{ij}^{\text{up}}), 6),$$

where $\epsilon_{ij}^{\text{low}}, \epsilon_{ij}^{\text{up}} \sim \text{Uniform}([0, \eta])$. We note that the distance should always be larger than the Van der Waal's radius which is 1.8 Åfor hydrogen.

Again, we first check whether the distributed algorithm can exactly recover the coordinates for the easy case of having a complete distance graph and noiseless distance measurements ($\eta = 0$). For this simulation we use a small protein 2MCE [31] with 317 atoms. Instead of just simulating distances between hydrogen atoms, we simulate all the $\binom{K}{2}$ distances for this molecule using the structure deposited in the PDB. Indeed exact recovery is demonstrated in Figure 2.8, with RMSD of 2.2e-6.

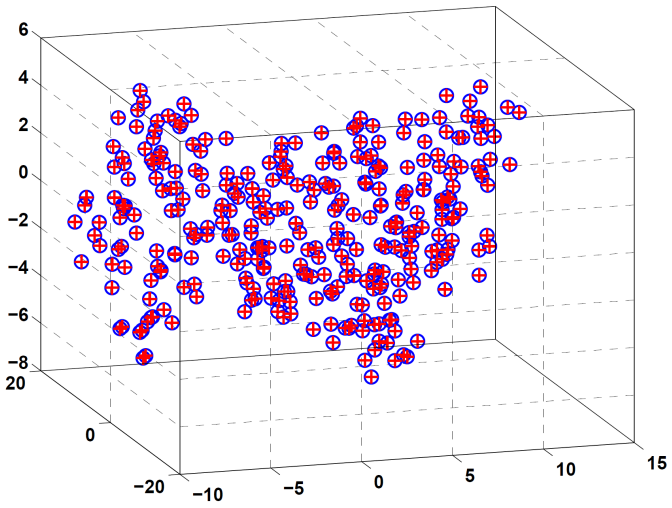


Figure 2.8: Comparison between the reconstruction of 2MCE from GRET (Blue) and the ground truth (Red) given precise and complete set of distance restraints.

We now run simulations with a more realistic setting on three larger molecules, 1GB1[65], 2LUQ[70] and 2MBL. We demonstrate that our algorithm is able to find molecule structure close to the ground truth from simulated distance bounds. For each noise level η we average the results over 10 experiments. In this simulation we only sample 30% of the hydrogen-hydrogen distances within 6 Å at random. We note that at $\eta = 0.8$, the average gap between simulated upper and lower distance bounds is similar to experimental data. We compare our results against the other two SDP based distributed algorithms ASAP and DISCO for solving the molecular conformation problem. Due to the global nature of the registration procedure, both GRET and ASAP have better RMSD than DISCO. Although GRET and ASAP have similar performances in simulated data, GRET outperforms ASAP in experimental data where there are fewer distance restraint, as shown in Table 2.2.

Protein	Distances	η	Gap	GRET	ASAP	DISCO
1GB1	NOE: 2400	0	0	0.40(0.20)	0.43(0.21)	0.46(0.22)
(855 atoms)	Covalent: 2900	0.4	1.51	0.70(0.50)	0.72(0.51)	0.84(0.55)
		0.8	2.60	0.85(0.70)	0.87(0.69)	1.22(0.92)
2LUQ	NOE:5000	0	0	1.23(0.74)	1.23(0.73)	1.25(0.74)
(1409 atoms)	Covalent: 4900	0.4	1.52	1.49(1.04)	1.49(1.01)	1.43(0.97)
		0.8	2.51	1.75(1.31)	1.76(1.33)	1.94(1.48)
2MBL	NOE:6000	0	0	0.59(0.42)	0.62(0.40)	0.78(0.51)
(1968 atoms)	Covalent:6600	0.4	1.53	1.46(1.04)	1.39(1.11)	1.83(1.47)
		0.8	2.66	2.00(1.44)	3.84(3.52)	2.09(1.90)

Table 2.1: Reconstruction error (in RMSD) for three molecules from simulated data. In the column with the header "Distances", we provide the number of simulated NOE restraints and covalent geometry constraints used in the simulation. In the column with the header "Gap", we report the average gap between the upper and lower bound of the distance restraints. The numbers in the columns with header "GRET","ASAP" and "DISCO" are the RMSD of the reconstructions produced by each of the methods, and the numbers in bracket are the RMSD for just the protein backbone. Each number is averaged over 10 different noise realizations.

Distances	GRET	ASAP	DISCO
NOE: 800 Covalent: 2900	2.23(1.61)	2.93(2.22)	2.46(1.80)

Table 2.2: Reconstruction error (RMSD) for the molecule 1GB1 (855 atoms). We compare the reconstructions with the solution NMR structure deposited in the PDB.

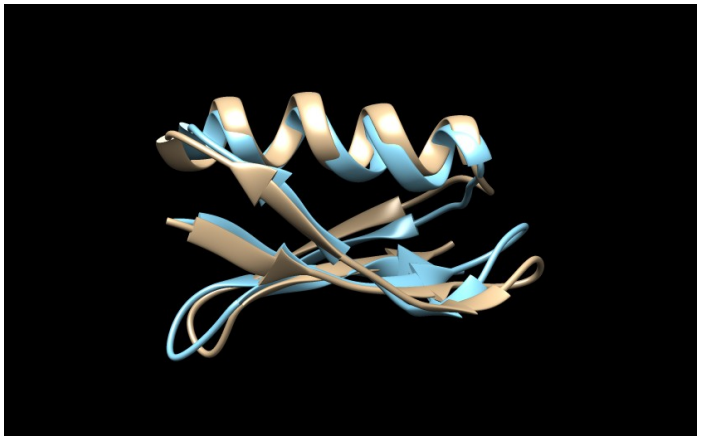


Figure 2.9: Comparison between the reconstruction from GRET (Blue) and the 1GB1 structure in the PDB (Gold).

2.7 Technical proofs

In this Section, we give the proof of Propositions 2.3.1 and 2.4.3, and Lemmas 2.4.4 and 2.4.5.

2.7.1 Proof of Proposition 2.3.1

We are done if we can show that there exists a bijection between the nullspace of C and that of C_0. To do so, we note that the associated quadratic forms can be expressed as

$$u^T C u = \min_{z \in \mathbb{R}^{1\times K+M}} \sum_{(k,i)\in E(\Gamma)} \|z e_{ki} - u_i^T x_k^{(i)}\|^2,$$

and

$$v^T C_0 v = \min_{z \in \mathbb{R}^{1\times K+M}} \sum_{(k,i)\in E(\Gamma)} \|z e_{ki} - v_i^T \bar{x}_k\|^2.$$

Here $u_1, \ldots, u_M$ are the $d \times 1$ blocks of the vector $u \in \mathbb{R}^{Md\times 1}$.

Now, it follows from (2.25) that there is a one-to-one correspondence between u and v, namely

$$u_i = \bar{O}_i v_i \qquad (1 \le i \le M),$$

such that $u^T C u = v^T C_0 v$. In other words, the null space of C is related to the null space of C_0 through an orthogonal transform, as was required to be shown.

2.7.2 Proof of Proposition 2.4.3

Without loss of generality, we assume that the smallest Euclidean ball that encloses the clean configuration $\{\bar{x}_1, \ldots, \bar{x}_K\}$ is centered at the origin, that is,

$$\|\bar{x}_k\| \leq R \qquad (1 \leq k \leq K). \tag{2.55}$$

Let B_0 be the matrix B in (2.12) computed from the clean measurements, i.e., from (2.33) with $\varepsilon = 0$. Let $B_0 + H$ be the same matrix obtained from (2.33) for some $\varepsilon > 0$.

Recall that $Z_0 = O_0 B_0 L^\dagger$ (by the centering assumption in (2.34)). Therefore,

$$\|Z^\star - \Theta Z_0\|_F = \|O^\star(B_0 + H)L^\dagger - \Theta O_0 B_0 L^\dagger\|_F = \|(O^\star - \Theta O_0)B_0 L^\dagger + O^\star H L^\dagger\|_F.$$

By triangle inequality,

$$\|Z^\star - \Theta Z_0\|_F \leq \|O^\star - \Theta O_0\|_F \|B_0 L^\dagger\|_F + \|O^\star H L^\dagger\|_F, \tag{2.56}$$

Now

$$\|B_0 L^\dagger\|_F \leq \|L^\dagger\|_{\mathrm{sp}} \|B_0\|_F = \frac{1}{\lambda_2(L)} \|B_0\|_F,$$

where $\lambda_2(L)$ is the smallest non-zero eigenvalue of L. On the other hand,

$$B_0 = \sum_{(k,i) \in E(\Gamma)} (e_i^M \otimes I_d) \bar{x}_k e_{ki}^T.$$

Using Cauchy-Schwarz and (2.55), we get

$$\begin{aligned}\|B_0\|_F^2 &= \sum_{(k,i)\in E(\Gamma)}\sum_{(l,j)\in E(\Gamma)} \mathrm{Tr}\Big(e_{ki}\bar{x}_k^T(e_i^M\otimes I_d)^T(e_j^M\otimes I_d)\bar{x}_l e_{lj}^T\Big)\\ &= \sum_{(k,i)\in E(\Gamma)}\sum_{(l,i)\in E(\Gamma)} \bar{x}_k^T\bar{x}_l\ e_{ki}^T e_{li}.\\ &\leq \sum_{(k,i)\in E(\Gamma)} 2R^2 + \sum_{(k,i)\in E(\Gamma)}\sum_{(l,i)\in E(\Gamma)} R^2.\end{aligned}$$

Therefore,

$$\|B_0L^\dagger\|_F \leq \lambda_2(L)^{-1}\sqrt{2+K}|E(\Gamma)|^{1/2}R. \tag{2.57}$$

As for the other term in (2.56), we can write

$$\|O^\star HL^\dagger\|_F \leq \|L^\dagger\|_{\mathrm{sp}}\|O^\star H\|_F = \lambda_2(L)^{-1}\|O^\star H\|_F.$$

Now

$$O^\star H = O^\star(B-B_0) = \sum_{(k,i)\in E(\Gamma)} O_i^\star \epsilon_k^{(i)} e_{ki}^T.$$

Therefore, using Cauchy-Schwarz, the orthonormality of the columns of $O_i^\star$'s, and the noise model (2.33), we get

$$\begin{aligned}\|O^\star H\|_F^2 &= \sum_{(k,i)\in E(\Gamma)}\sum_{(l,j)\in E(\Gamma)} (O_i^\star\epsilon_k^{(i)})^T(O_j^\star\epsilon_{l,j})e_{ki}^T e_{li}\\ &\leq \sum_{(k,i)\in E(\Gamma)} 2\varepsilon^2 + \sum_{(k,i)\in E(\Gamma)}\sum_{(l,i)\in E(\Gamma)} \varepsilon^2 + \sum_{(k,i)\in E(\Gamma)}\sum_{(k,j)\in E(\Gamma)} \varepsilon^2.\end{aligned}$$

This gives us

$$\|O^\star HL^\dagger\|_F \leq \sqrt{2+K+M}|E(\Gamma)|^{1/2}\lambda_2(L)^{-1}\varepsilon. \tag{2.58}$$

Combining (2.56),(2.57), and (2.58), we get the desired estimate.

2.7.3 Proof of Lemma 2.4.4

The proof is mainly based on the observation that if u and v are unit vectors and $0 \le u^T v \le 1$, then

$$\|u - v\| \le \|uu^T - vv^T\|_F. \tag{2.59}$$

Indeed,

$$\|uu^T - vv^T\|_F^2 = \mathrm{Tr}\left(uu^T + vv^T - 2(u^T v)^2\right) \ge \mathrm{Tr}(uu^T + vv^T - 2u^T v) = \|u - v\|^2.$$

To use this result in the present setting, we use the theory of principal angles [16, Ch. 7.1]. This tells us that, for the orthonormal systems $\{u_1, \ldots, u_d\}$ and $\{s_1, \ldots, s_d\}$, we can find $\Omega_1, \Omega_2 \in \mathbb{O}(Md)$ such that

1. $\Omega_1[u_1 \cdots u_d] = [u_1 \cdots u_d]\Theta_1^T$ where $\Theta_1 \in \mathbb{O}(d)$,
2. $\Omega_2[s_1 \cdots s_d] = [s_1 \cdots s_d]\Theta_2^T$ where $\Theta_2 \in \mathbb{O}(d)$,
3. $(\Omega_1 s_i)^T(\Omega_2 u_j) = 0$ for $i \ne j$, and $0 \le (\Omega_1 s_i)^T(\Omega_2 u_i) \le 1$ for $1 \le i \le d$.

Here Θ_1 and Θ_2 are the orthogonal transforms that map $\{u_1, \ldots, u_d\}$ and $\{s_1, \ldots, s_d\}$ into the corresponding principal vectors.

Using properties 1 and 2 and the fact[3] that $\alpha_i \le M$, we can write

$$\sqrt{M}\ \|\Theta_1 W^\star - \Theta_2 O_0\|_F \le \|\Omega_1[\alpha_1 u_1 \cdots \alpha_d u_d] - M\Omega_2[s_1 \cdots s_d]\|_F + \Big[\sum_{i=1}^{d}(M - \alpha_i)^2\Big]^{1/2}.$$

[3] To see why the eigenvalues of $G^\star$ are at most M, note that by the SDP constraints, for every block G_{ij},

$$u^T G_{ij} v \le (\|u\|^2 + \|v\|^2)/2 \quad (u, v \in \mathbb{R}^d).$$

Let $x = (x_1, \ldots, x_M)$ where each $x_i \in \mathbb{R}^d$. Then

$$x^T G x = \sum_{i,j} x_i^T G_{ij} x_j \le \sum_{i,j} (\|x_i\|^2 + \|x_j\|^2)/2 = M\|x\|^2.$$

Moreover, by triangle inequality,

$$\|\Omega_1[\alpha_1 u_1 \cdots \alpha_d u_d] - M\Omega_2[s_1 \cdots s_d]\|_F \leq M\|\Omega_1[u_1 \cdots u_d] - \Omega_2[s_1 \cdots s_d]\|_F + \Big[\sum_{i=1}^{d}(M-\alpha_i)^2\Big]^{1/2}.$$

Therefore,

$$\sqrt{M}\ \|\Theta_1 W^\star - \Theta_2 O_0\|_F \leq M\|\Omega_1[u_1 \cdots u_d] - \Omega_2[s_1 \cdots s_d]\|_F + 2\Big[\sum_{i=1}^{d}(M-\alpha_i)^2\Big]^{1/2}.$$

Now, using (2.59) and the principal angle property 3, we get

$$\|\Omega_1[u_1 \cdots u_d] - \Omega_2[s_1 \cdots s_d]\|_F \leq \|\sum_{i=1}^{d}\Omega_1 u_i(\Omega_1 u_i)^T - \sum_{i=1}^{d}\Omega_2 s_i(\Omega_2 s_i)^T\|_F.$$

Moreover, using triangle inequality and properties 1 and 2, we have

$$M\|\sum_{i=1}^{d}\Omega_1 u_i(\Omega_1 u_i)^T - \sum_{i=1}^{d}\Omega_2 s_i(\Omega_2 s_i)^T\|_F \leq \|W^{\star T}W^\star - G_0\|_F + \Big[\sum_{i=1}^{d}(M-\alpha_i)^2\Big]^{1/2}.$$

Finally, note that by Lemma 2.4.6,

$$\Big[\sum_{i=1}^{d}(M-\alpha_i)^2\Big]^{1/2} \leq \|W^{\star T}W^\star - G_0\|_F. \tag{2.60}$$

Combining the above relations, and setting $\Theta = \Theta_1^T\Theta_2$, we arrive at Lemma 2.4.4.

2.7.4 Proof of Lemma 2.4.5

This is done by adapting the following result by Li [98]: If A, B are square and non-singular, and if $\mathscr{R}(A)$ and $\mathscr{R}(B)$ are their orthogonal rounding (obtained from their polar decompositions [74]), then

$$\|\mathscr{R}(A) - \mathscr{R}(B)\|_F \leq \frac{2}{\sigma_{\min}(A) + \sigma_{\min}(B)}\|A - B\|_F. \tag{2.61}$$

We recall that if $A = U\Sigma V^T$ is the SVD of A, then $\mathscr{R}(A) = UV^T$.

Note that it is possible that some of the blocks of $W^\star$ are singular, for which the above result does not hold. However, the number of such blocks can be controlled by the global error. More precisely, let $\mathscr{B} \subset \{1, 2, \ldots, M\}$ be the index set such that, for $i \in \mathscr{B}$, $\|W_i^\star - \Theta\|_F \geq \beta$. Then

$$\|W^\star - \Theta O_0\|_F^2 \geq \sum_{i \in \mathscr{B}} \|W_i^\star - \Theta\|_F^2 = |\mathscr{B}|\beta^2.$$

This gives a bound on the size of $\mathscr{B}$. In particular, the rounding error for this set can trivially be bounded as

$$\sum_{i \in \mathscr{B}} \|O_i^\star - \Theta\|_F^2 \leq \sum_{i \in \mathscr{B}} 2d = \frac{2d}{\beta^2} \|W^\star - \Theta O_0\|_F^2. \tag{2.62}$$

On the other hand, we known that, for $i \in \mathscr{B}^c$, $\|W_i^\star - \Theta\|_F < \beta$. From Lemma 2.4.6, it follows that

$$|1 - \sigma_{\min}(W_i^\star)| \leq \|W_i^\star - \Theta\|_{\text{sp}} < \beta.$$

Fix $\beta \leq 1$. Then $\sigma_{\min}(W_i^\star) > 1 - \beta$, and we have from (2.61),

$$\|O_i^\star - \Theta\|_F \leq \frac{2}{2-\beta} \|W_i^\star - \Theta\|_F \qquad (i \in \mathscr{B}^c) \tag{2.63}$$

Fixing $\beta = 1/\sqrt{2}$ and combining (2.62) and (2.63), we get the desired bound.

Chapter 3

RDC-based method for protein structuring

In NMR spectroscopy, for large molecules the extraction of NOE restraints through resonance assignment is difficult and often leads to missing, ambiguous, or incorrect NOE distance measurements. Hence the inverse problem of positioning from distance constraints alone, also known as the distance geometry problem, can be challenging and even ill-posed [165]. In this case, the coordinates of the atoms obtained from the NOE restraints may not have satisfactory accuracy. Thus in this chapter, we focus on using RDC measurements

$$r_{nm} = \frac{(x_n - x_m)^T S (x_n - x_m)}{d_{nm}^2}, \tag{3.1}$$

to obtain atom coordinates with high accuracy. Throughout this chapter we assume that S can be estimated a-priori [101, 172] and our goal is to determine the atom positions given S.

The NOE and RDC constraints we described so far are in terms of the Cartesian coordinates of the atoms. However, a protein can be viewed as an articulated structure which is composed of rigid planes and bodies that are chained together via hinges [69]. As we will see in later sections of the chapter, the atom coordinates can therefore be

expressed in terms of rotations associated with the rigid units. The determination of the rotations from RDC and NOE then provides the protein structure. In a broader context, our solution to the protein structuring problem presents a general strategy for determining the pose of an articulated structure, a common problem that arises in robotics and computer vision [55, 6]. The way we model the articulated structure from rotation matrices results in a cost function and constraints that are separable in the rotations, which in turn facilitates subsequent optimization. We also strengthen the convex relaxation proposed in [7], which originally intends to minimize quadratic functions involving orthogonal matrices, in order to deal with special orthogonal transformations. This is particularly meaningful in practical applications as rigid units in an articulated structure do not usually undergo a reflection. As shown by our numerical experiments, the additional constraints specific to the special orthogonal group greatly enhance noise stability.

The rest of the chapter is organized as follows. In Section 3.2, we formulate the problem of backbone structure determination from RDC and NOE as a problem of finding the pose of an articulated structure. In Section 3.3, we describe a semidefinate program (SDP) for solving optimization problems involving quadratic functions of rotation and we apply such SDP in Section 3.4 to determine the pose of an articulated structure. In Section 3.5, we propose an alternate SDP to find the relative translations between fragments, when estimating the full protein structure directly is not possible. In Section 3.6, we present the numerical results with synthetic data and also for experimental data of ubiquitin (PDB ID: 1D3Z). In Section 3.8, we introduce the Cramér-Rao lower bound for the structure determination problem from RDC.

3.1 Notation

We use I_d to denote the identity matrix of size $d \times d$. We use A_i to denote the i-th column of a matrix A. We use vec(A) to denote the vectorization of a matrix A, and mat(A) to denote the inverse procedure. In this chapter we only use the mat($\cdot$) operation to form a 3×3 matrix from a column vector in $\mathbb{R}^9$. We denote the trace of a square matrix A by Tr(A). The Kronecker product between matrices A and B is denoted by $A \otimes B$. The all-ones vector is denoted by $\mathbf{1}$ (the dimension should be obvious from the context). The i-th canonical basis vector is denoted as e_i.

3.2 Problem Formulation

3.2.1 Protein backbone as articulated structure

An articulated structure is a chain of rigid units where one unit is "chained" together with the next unit with non-overlapping joints (Figure 3.1a). When there is a joint between two consecutive units, the relative translation is fixed but not the relative rotation. If there are two non-overlapping joints between two consecutive units, there is only one undetermined degree of freedom corresponding to a rotation around the axis defined by the two joints. This structure is also referred to as the *body-hinge* framework [158] in rigidity theory. Let an articulated structure be composed of K points residing in M rigid units. For such structure, we define a set of points $\{J_i\}_{i=1}^{M}$ as the joints between the units where $J_i \in \{1, \ldots, K\}$. The i-th unit is joined to the $(i-1)$-th unit at J_i. Since the coordinates in each unit are known a-priori up to a rigid transformation, we then use $x_k^{(i)}$ to denote the location of point k in the local coordinate system of the i-th rigid unit. Notice that due to the rigid motion ambiguity, a Euclidean transform needs to be applied to each of the local coordinates $x_k^{(i)}$ for each i in order to form the global structure.

Let $\zeta_k^{(i)}$ be the global coordinate of point k in the i-th unit. For an articulated structure,

it is possible to represent the global coordinates $\zeta_k^{(i)}$ using the rotations $R_i, i = 1, \ldots, M$ associated with the M rigid units. For $i = 1$, we let

$$\zeta_k^{(1)} = R_1(x_k^{(1)} - x_{J_1}^{(1)}) + t \tag{3.2}$$

which amounts to orienting the first rigid unit with R_1 and adding a translation so that $\zeta_{J_1}^{(1)}$ are placed at $t \in \mathbb{R}^3$. The coordinates for the $i = 2$ rigid unit can be obtained as

$$\zeta_k^{(2)} = R_2(x_k^{(2)} - x_{J_2}^{(2)}) + \zeta_{J_2}^{(1)}. \tag{3.3}$$

The above operations ensure that the $i = 2$ rigid unit is jointed to the $i = 1$ rigid unit at joint J_2, since one can verify that $\zeta_{J_2}^{(2)} = \zeta_{J_2}^{(1)}$. The same reasoning implies that in general the recursive relationship

$$\zeta_k^{(i)} = R_i(x_k^{(i)} - x_{J_i}^{(i)}) + \zeta_{J_i}^{(i-1)} \tag{3.4}$$

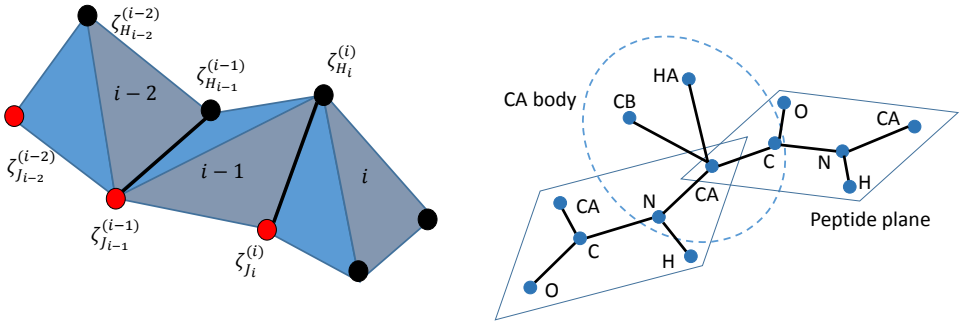
should hold. Applying induction to (3.4) results

$$\zeta_k^{(i)} = R_i(x_k^{(i)} - x_{J_i}^{(i)}) + \sum_{s=1}^{i-1} R_s(x_{J_{s+1}}^{(s)} - x_{J_s}^{(s)}) + t. \tag{3.5}$$

The coordinate of each atom is thus expressed as a linear combination of the rotations R_i's and a global translation t. As mentioned previously, when there are hinges in the articulated structure the rotations have fewer degrees of freedom. To incorporate the hinges, we define another set of joints $\{H_i\}_{i=1}^M$ where $\{H_i\}_{i=1}^M \cap \{J_i\}_{i=1}^M = \emptyset$. Let $v_{kl}^{(i)}$ be the unit vector between the pair of points (k, l) in the frame of the i-th rigid unit. To ensure two consecutive rigid bodies stay chained together by a hinge, R_i's should satisfy the hinge constraints

$$R_i v_{H_i J_i}^{(i)} = R_{i-1} v_{H_i J_i}^{(i-1)}, \quad i = 2, \ldots, M. \tag{3.6}$$

Using the above framework, we can reduce the problem of finding atomic coordinates of a protein backbone into a problem of finding the special orthogonal transforms. This is because the protein backbone can be modeled as an articulated structure composed of peptide planes and CA-bodies. As depicted in Figure 3.1b, a peptide plane is a 2D rigid plane consisting atoms from two consecutive amino acids: CA, C, O from one amino acid and H, N, CA from the next amino acid. The CA-body is a 3D rigid body consisting of five atoms CA, N, C, HA and CB all coming from one amino acid. The bonds (N, CA), (C, CA) act like hinges between the rigid units.



(a) (b)

Figure 3.1: (a): Example of an articulated structure with joints with indices J_i's (Red dots) and H_i's. The hinges are represented by black bars in the figure. (b): Protein backbone consists of peptide planes and CA bodies. These rigid units are chained together at the bonds (N, CA) and (C,CA).

3.2.2 RDC data

In the setting of calculating protein structure, the RDC measurements described in (3.1) can be used to constrain the rotation for each rigid unit. Within each rigid unit, in principle all pairs of isotope-labeled atoms except those involving oxygen, O, can give rise to RDC, although in practice only a subset of these pairs have their RDC measured. Suppose N Saupe tensors for the protein in N different alignment media have been predetermined. In the j-th alignment media, the RDC measurements for the i-th rigid unit between the pair of atoms (n, m), denoted $r_{nm}^{(j)}$, can be modeled in the following

way:

$$r_{nm}^{(j)} = {v_{nm}^{(i)}}^T R_i^T S^{(j)} R_i v_{nm}^{(i)}, \quad (n,m) \in E_{\mathrm{RDC}i}, \\ i = 1,\ldots,M, \quad j = 1,\ldots,N. \tag{3.7}$$

The set $E_{\mathrm{RDC}i}$ is the set of edges that give rise to RDC in the i-th rigid unit, and $S^{(j)}$ denotes the Saupe tensor in alignment media j. The orientation of the peptide planes and CA-bodies can be obtained by solving equation (3.7) subject to the hinge constraint (3.6). Due to experimental errors in measuring the RDC, (3.7) is only satisfied approximately, and orientations can be estimated by minimizing the following cost

$$\sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{(n,m)\in E_{\mathrm{RDC}i}} |{v_{nm}^{(i)}}^T R_i^T S^{(j)} R_i v_{nm}^{(i)} - r_{nm}^{(j)}|^p \tag{3.8}$$

subject to (3.6). In the cost function (3.8) each bond is counted once, including bonds that lie in both the peptide plane and the CA-body (e.g., bond $(\mathrm{C}-\mathrm{CA})$). The choice of the parameter p depends on the specific noise model, and typical choices are $p = 2$ (least squares) and $p = 1$ (least unsquared deviations). We show in Section 3.8.2, the minimization of (3.8) with $p = 2$ corresponds to a maximum likelihood estimation when the noise on RDC is Gaussian. If robustness to outlier type noise is required, $p = 1$ can be used instead. The difficulty of minimizing target function (3.8) lies in the non-convex nature of both the cost and domain. Therefore, RDC measurements are typically used when refining an existing, high quality structure derived from solving the distance geometry problem from NOE or from homology modeling [33].

3.2.3 NOE data

We now rewrite the distance constraints in (1.4) in terms of the rotations. Instead of working with bounds on distances, we use bounds on squared distances, for reasons

that will become apparent in Section 3.3. Assuming $i > j$, from (3.5) we have

$$\|\zeta_m^{(i)} - \zeta_n^{(j)}\|_2^2 \;=\; \|R_i(x_m^{(i)} - x_{J_i}^{(i)}) - R_j(x_n^{(j)} - x_{J_j}^{(j)}) + \sum_{s=j+1}^{i-1} R_s(x_{J_{s+1}}^{(s)} - x_{J_s}^{(s)})\|_2^2. \quad (3.9)$$

In this way, we write squared distances between two atoms, necessary for expressing NOE measurements, as quadratic functions of R_i's. To satisfy the constraint (1.4), we can minimize

$$\max((d_{mn}^{\text{low}})^2 - \|\zeta_m^{(i)} - \zeta_n^{(j)}\|_2^2, 0)^p + \max(\|\zeta_m^{(i)} - \zeta_n^{(j)}\|_2^2 - (d_{mn}^{\text{up}})^2, 0)^p \quad (3.10)$$

where the choice of p again depends on the noise model. In practice, the NOE measurements for the backbone atoms are more reliable and can also be treated as relatively hard constraints.

3.3 Quadratic problem on $\mathbb{O}(3)$ and $\mathbb{SO}(3)$

In this section, we introduce a novel convex relaxation to optimization problems of the form

$$\min_R f(\text{vec}(R)\text{vec}(R)^T) \quad \text{such that } R \in \mathbb{SO}(3) \quad (3.11)$$

where f is a convex function, upon which our method for estimating pose of an articulated structure relies. We note that a convex relaxation has been proposed previously in [7] to a close relative of problem (3.11), namely

$$\min_R f(\text{vec}(R)\text{vec}(R)^T) \quad \text{such that } R^TR = I_3,\ RR^T = I_3 \quad (3.12)$$

where R belongs to the orthogonal group. However, since we consider the group of $\mathbb{SO}(3)$ instead of the orthogonal group we can further strengthen the relaxation in [7] by relating matrices in $\mathbb{SO}(3)$ to their quaternion representation. Before proceeding we

introduce some notations. The linear operator $\mathscr{R} : \mathbb{R}^{9\times 9} \to \mathbb{R}^{3\times 3}$ is defined as

$$\mathscr{R}(X)(i,j) = \mathrm{Tr}(X_{ij}) \tag{3.13}$$

for any $X \in \mathbb{R}^{9\times 9}$ where X_{ij} denotes the (i,j)-th 3×3 block in X. The operator $\mathscr{R}$ enables writing the product

$$A^T B = \mathscr{R}(\mathrm{vec}(A)\mathrm{vec}(B)^T). \tag{3.14}$$

for any two 3×3 matrices A and B. The linear operator $\mathscr{L} : \mathbb{R}^{9\times 9} \to \mathbb{R}^{3\times 3}$ is defined as

$$\mathscr{L}(X) = \sum_{i=1}^{3} X_{ii}. \tag{3.15}$$

Notice that for any 3×3 matrices A, B,

$$AB^T = \mathscr{L}(\mathrm{vec}(A)\mathrm{vec}(B)^T). \tag{3.16}$$

3.3.1 Convex relaxation: quadratic problem on $\mathbb{O}(3)$

We first discuss the instance of solving equation (3.12) where we only consider variables in the orthogonal group $\mathbb{O}(3)$. In order to derive a relaxation of (3.12), we define a new variable

$$Y = \mathrm{vec}(R)\mathrm{vec}(R)^T \tag{3.17}$$

that consists of all degree 2 monomials of the elements of R. To enforce orthogonality, we add the constraints

$$\begin{aligned} I_3 = RR^T &= \mathscr{L}(\mathrm{vec}(R)\mathrm{vec}(R)^T) = \mathscr{L}(Y) \\ I_3 = R^T R &= \mathscr{R}(\mathrm{vec}(R)\mathrm{vec}(R)^T) = \mathscr{R}(Y) \end{aligned} \tag{3.18}$$

Although at this point the two constraints are redundant as $R^TR = I_3$ if and only if $RR^T = I_3$, its usefulness will be apparent when we apply convex relaxation. Using the newly defined variable Y, we first consider rewriting the problem (3.12) as

$$\begin{aligned} & \min_{Y,R} f(Y) \\ \text{s.t.} \quad & \mathscr{L}(Y) = \mathscr{R}(Y) = I_3, \\ & Y = \text{vec}(R)\text{vec}(R)^T \end{aligned} \tag{3.19}$$

The last constraint is equivalent to $Y \succeq 0$ and $\text{rank}(Y) = 1$. We then drop the rank constraint on Y and obtain the following SDP relaxation

$$\begin{aligned} & \min_{Y} f(Y) \\ \text{s.t.} \quad & \mathscr{L}(Y) = \mathscr{R}(Y) = I_3, \\ & Y \succeq 0. \end{aligned} \tag{3.20}$$

Semidefinite relaxation of this type was presented in [7]. It was further shown that for $f(Y) = \text{Tr}((A \otimes B)Y)$ where A, B are general symmetric matrices, the non-convex problem in (3.19) can be solved exactly via this type of relaxation. Notice that if $\text{rank}(Y) = 1$ such that $Y = \text{vec}(R)\text{vec}(R)^T$ for some $R \in \mathbb{R}^{3\times 3}$, the constraints $\mathscr{L}(Y) = \mathscr{R}(Y) = I_3$ are redundant. This is because $I_3 = \mathscr{L}(Y) = RR^T$ implies R^T is the inverse of R leading to $\mathscr{R}(Y) = R^TR = I_3$. This argument does not work if $Y \neq \text{vec}(R)\text{vec}(R)^T$ for some $R \in \mathbb{R}^{3\times 3}$ hence $\mathscr{L}(Y) \neq RR^T$ and $\mathscr{R}(Y) \neq R^TR$. In fact for the following Y with $\text{rank}(Y) = 3$ where

$$Y_{ii} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \; i = 1, 2, 3, \text{ and } Y_{ij} = 0 \text{ for } i \neq j,$$

$Y \succeq 0$ satisfies $\mathscr{L}(Y) = I_3$ but $\mathscr{R}(Y) \neq I_3$. Therefore after the rank relaxation both the constraints in (3.18) are needed and they are not redundant.

3.3.2 Convex relaxation: quadratic problem on $\mathbb{SO}(3)$

For physical problems, often we can further reduce the search space for R from $\mathbb{O}(3)$ to $\mathbb{SO}(3)$ due to chirality constraints. It would be beneficial if we can include the constraint $\det(R) = 1$. We have seen that the orthogonality of R can be enforced through linear constraints in (3.20) due to the fact that any degree 2 polynomial in R can be expressed as a linear function of $Y = \text{vec}(R)\text{vec}(R)^T$. However, the determinant constraint involves a degree 3 polynomial in the entries of R hence it cannot be expressed by the variables in (3.20). We therefore enforce chirality constraints by relating the columns of R through the cross products. Let

$$\text{Cross}(A) := \begin{bmatrix} A(2,3) - A(3,2) \\ A(3,1) - A(1,3) \\ A(1,2) - A(2,1) \end{bmatrix} \tag{3.21}$$

for any $A \in \mathbb{R}^{3\times 3}$. For two vectors $v_1, v_2 \in \mathbb{R}^3$, $\text{Cross}(v_1 v_2^T) = v_1 \times v_2$ where $v_1 \times v_2$ denotes the cross products between v_1 and v_2. For a rotation matrix $R \in \mathbb{SO}(3)$, the following constraints

$$R_1 = R_2 \times R_3 = \text{Cross}(Y_{23}), \quad R_2 = R_3 \times R_1 = \text{Cross}(Y_{31}), \quad R_3 = R_1 \times R_2 = \text{Cross}(Y_{12}) \tag{3.22}$$

specify the "handed-ness" of the coordinate frame established by $R = [R_1, R_2, R_3]$. Here $Y = \text{vec}(R)\text{vec}(R)^T$ and Y_{ij} is the (i,j)-th 3×3 block of Y. Let

$$\mathscr{X}(Y) := \begin{bmatrix} \text{Cross}(Y_{23}) & \text{Cross}(Y_{31}) & \text{Cross}(Y_{12}) \end{bmatrix},$$

problem in (3.11) can be written equivalently as

$$\begin{aligned} & \min_{Y,R} f(Y) \\ \text{s.t.} \quad & \mathscr{L}(Y) = \mathscr{R}(Y) = I_3, \end{aligned}$$

$$
\begin{aligned}
& Y = \text{vec}(R)\text{vec}(R)^T, \\
& R = \mathscr{X}(Y)
\end{aligned} \tag{3.23}
$$

Since the constraint $Y = \text{vec}(R)\text{vec}(R)^T$ is not convex, we replace it with $Y \succeq \text{vec}(R)\text{vec}(R)^T$, which results in a convex relaxation for quadratic problems on $\mathbb{SO}(3)$

$$
\begin{aligned}
& \min_{Y,R} f(Y) \\
\text{s.t.} \quad & \mathscr{L}(Y) = \mathscr{R}(Y) = I_3, \\
& Y \succeq \text{vec}(R)\text{vec}(R)^T, \\
& R = \mathscr{X}(Y)
\end{aligned} \tag{3.24}
$$

Interestingly in (3.24), R is in the convex hull of the rotation matrices. This can be seen by relating the elements in $\mathbb{SO}(3)$ to their unit quaternion representations, as shown in the appendix in Section 3.8.1.

We note that in [44], a similar convex relaxation using the cross products is proposed to optimize quadratic functions with their domain being the Stiefel manifold

$$
\{Q \in \mathbb{R}^{3\times 2} \mid Q^T Q = I_2\}.
$$

As in (3.19), such an optimization problem is convex in the PSD variable

$$
X = \text{vec}(Q)\text{vec}(Q)^T \succeq 0 \tag{3.25}
$$

if the rank-1 constraint on X is to be dropped. The orthogonality of the columns of Q can be enforced through placing linear constraints on X, i.e.

$$
\text{Tr}(X_{ij}) = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases} \tag{3.26}
$$

where X_{ij} denotes the (i,j)-th 3×3 block of X. In [44], an additional vector

$$Q_3 := Q_1 \times Q_2 = \text{Cross}(X_{12}) \tag{3.27}$$

is employed to further tighten this convex relaxation. Since the rows of the matrix $[Q_1, Q_2, Q_3]$ are

$$\begin{bmatrix} Q_1 & Q_2 & Q_3 \end{bmatrix} \begin{bmatrix} Q_1 & Q_2, & Q_3 \end{bmatrix}^T \preceq I_3, \tag{3.28}$$

implying the following convex constraint

$$X_{11} + X_{22} + Q_3 Q_3^T \preceq I_3. \tag{3.29}$$

This mimics the first constraint in (3.18) when dealing with orthogonal matrices. However, equality cannot be placed on equation (3.29) since this introduces non-convexity.

3.4 Quadratic problem of articulated structures

In this section, we propose two convex relaxations for finding the pose of an articulated structure. In this case, we need to solve for the rotation of each of the M rigid units subject to the hinge constraints in (3.6). We first define variables

$$R = [R_1, \ldots, R_M] \in \mathbb{SO}(3)^M$$

and

$$Y = \text{vec}(R)\text{vec}(R)^T. \tag{3.30}$$

For convenience of indexing, in this section we view Y as a $M \times M$ block matrix where $Y_{ij} = \text{vec}(R_i)\text{vec}(R_j)^T$. It is important to define such a matrix Y since the measurements involve quadratic functions of rotation matrices.

If the rigid units are not chained together, each R_i can be solved for via the convex relaxation proposed in (3.24). However, in an articulated structure the rigid units are not independent of each other but related via (3.6)

$$R_i v^{(i)}_{H_i J_i} = R_{i-1} v^{(i-1)}_{H_i J_i}, \quad i = 2, \ldots, M \tag{3.31}$$

which are linear constraints between R_i and R_{i-1}. Therefore all rotations have to be optimized jointly. We now introduce a few redundant constraints. Equation (3.6) leads to constraints on Y:

$$v^{(i-1)^T}_{J_i H_i} R^T_{i-1} e^T_k e_l R_{i-1} v^{(i-1)}_{J_i H_i} = v^{(i)\ T}_{J_i H_i} R^T_i e^T_k e_l R_i v^{(i)}_{J_i H_i} \quad \forall\ k, l = 1, 2, 3, \tag{3.32}$$

where e_k's are the canonical basis vectors in $\mathbb{R}^3$. Writing the constraints using Y we get

$$\mathrm{Tr}((v^{(i-1)}_{J_i H_i} v^{(i-1)^T}_{J_i H_i} \otimes e_k e_l) Y_{(i-1)(i-1)}) = \mathrm{Tr}((v^{(i)}_{J_i H_i} v^{(i)\ T}_{J_i H_i} \otimes e_k e_l) Y_{ii}). \tag{3.33}$$

In the same spirit, another set of constraints

$$v^{(i-1)}_{J_i H_i} = R^T_{i-1} R_i v^{(i)}_{J_i H_i}$$

can be encoded as

$$v^{(i-1)}_{J_i H_i} = \mathscr{R}(Y_{(i-1)i}) v^{(i)}_{J_i H_i}. \tag{3.34}$$

The redundant constraints (3.33) and (3.34) will no longer be redundant when $Y = \mathrm{vec}(R)\mathrm{vec}(R)^T$ is relaxed to $Y \succeq \mathrm{vec}(R)\mathrm{vec}(R)^T$.

Now, based on the convex relaxation (3.24) for the problem involving a single rotation, together with the hinge constraints equations (3.6),(3.33) and (3.34), we propose the following convex relaxation to solve for the rotations for an articulated

structure:

$$\begin{aligned}
\text{(P1)} \qquad & \min_{Y,R} f(Y) && (3.35)\\
\text{s.t.} \quad & Y \succeq \text{vec}(R)\text{vec}(R)^T && (3.36)\\
& \mathscr{L}(Y_{ii}) = \mathscr{R}(Y_{ii}) = I_3,\ i \in [1,M], && (3.37)\\
& R_i = \mathscr{X}(Y_{ii}),\ i \in [1,M] && (3.38)\\
& R_{i-1} v_{J_i H_i}^{(i-1)} = R_i v_{J_i H_i}^{(i)},\ i \in [2,M], && (3.39)\\
& v_{J_i H_i}^{(i-1)} = \mathscr{R}(Y_{(i-1)i}) v_{J_i H_i}^{(i)},\ i \in [2,M], && (3.40)\\
& \text{Tr}((v_{J_i H_i}^{(i-1)} v_{J_i H_i}^{(i-1)^T} \otimes e_k e_l) Y_{(i-1)(i-1)}) \\
& \quad = \text{Tr}((v_{J_i H_i}^{(i)} v_{J_i H_i}^{(i)^T} \otimes e_k e_l) Y_{ii}),\ k,l \in [1,3],\ i \in [2,M] && (3.41)
\end{aligned}$$

Here f is a convex function determined by the measurements. As before, the relaxation is obtained by changing $Y = \text{vec}(R)\text{vec}(R)^T$ to (3.36).

The SDP problem (P1) involves a PSD variable of size $(9M+1) \times (9M+1)$. In applications where the convex cost of (P1) can be decomposed as

$$f(Y) = \sum_{i=1}^{M} f_i(Y_{ii}), \tag{3.42}$$

i.e. each term in the cost involves a single rotation, the size of the variable used in (P1) can be further reduced. In this case, we propose the following size-reduced convex

relaxation

$$\text{(P2)} \qquad \min_{Y^{(i)},R_i \succeq 0} \sum_{i=1}^{M} f_i(Y^{(i)}) \tag{3.43}$$

$$\text{s.t.} \quad Y^{(i)} \succeq \text{vec}(R_i)\text{vec}(R_i)^T, \tag{3.44}$$

$$\mathscr{L}(Y^{(i)}) = \mathscr{R}(Y^{(i)}) = I_3,\ i \in [1, M], \tag{3.45}$$

$$R_i = \mathscr{X}(Y^{(i)}),\ i \in [1, M] \tag{3.46}$$

$$R_{i-1} v_{J_i H_i}^{(i-1)} = R_i v_{J_i H_i}^{(i)},\ i \in [2, M], \tag{3.47}$$

$$\begin{aligned} &\text{Tr}((v_{J_i H_i}^{(i-1)} v_{J_i H_i}^{(i-1)^T} \otimes e_k e_l)\mathscr{R}(Y^{(i)})) \\ &= \text{Tr}((v_{J_i H_i}^{(i)} v_{J_i H_i}^{(i)^T} \otimes e_k e_l)\mathscr{R}(Y^{(i)})),\ k, l \in [1, 3],\ i \in [2, M]. \end{aligned} \tag{3.48}$$

All the constraints of (P2) are implied by the constraints in (P1) except (3.40). Notice that if the constraint (3.40) is not included in (P1), then (P2) and (P1) are in fact equivalent under the assumption that the cost function satisfies (3.42). From a solution $Y^{(i)^\star}, R_i^\star$ of (P2), a solution $Y^\star$ in (P1) can be obtained by simply setting $Y_{ii}^\star = Y^{(i)^\star}$ and $Y_{ij}^\star = 0$ for $i \neq j$, with the same $R_i^\star$ from (P2).

We pause here for a remark about the convex relaxation in (P1). If the function f only depends on $R_i^T R_j$ (which is the case when only NOE measurements are provided for protein structural determination), it suffices to use a classic SDP proposed for rotation synchronization problem involving a $3M \times 3M$ rank-3 Gram matrix [134, 41]

$$G := \begin{bmatrix} R_1^T \\ \vdots \\ R_M^T \end{bmatrix} \begin{bmatrix} R_1 & \dots & R_M \end{bmatrix}. \tag{3.49}$$

Define the (i, j)-th 3×3 block of G as G_{ij}, we can minimize $f(G)$ (f is convex) using the Max-Cut type SDP relaxation [56]

$$\min_G f(G)$$

$$
\begin{aligned}
\text{s.t.}\quad & G_{ii} = I_3, \\
& G \succeq 0, \\
& v_{J_i H_i}^{(i-1)} = G_{(i-1)i} v_{J_i H_i}^{(i)},\ i \in [2, M], \\
& \operatorname{rank}(G) = 3 \text{ (relaxed)}.
\end{aligned} \tag{3.50}
$$

In this context of f arising solely from NOE restraints, this program can be used to solve the distance geometry problem. In this case, (P1) is an overly-relaxed convex relaxation as there are many more variables in (P1) compare to (3.50), with the same number of measurements. In the presence of both RDC and NOE constraints, (P1) is needed instead since the cost depends on individual columns of R_i. We note that the problem (3.50) is embedded in (P1). More precisely, letting $G_{ij} := \mathscr{R}(Y_{ij})$, the constraints in (3.50) are implied by the constraints in (P1). While it is obvious to see this for the linear constraints in (3.50), to see the PSD-ness of G, first let $\mathscr{R}^*$ be the adjoint operator of $\mathscr{R}$ defined through

$$
\operatorname{Tr}(B^T \mathscr{R}(A)) = \operatorname{Tr}(A^T \mathscr{R}^*(B)) \quad \text{for any } A \in \mathbb{R}^{9\times 9},\ B \in \mathbb{R}^{3\times 3}.
$$

Then

$$
\mathscr{R}^*(B) = B \otimes I_3.
$$

$G \succeq 0$ follows from the fact that for any $x \in \mathbb{R}^{3M}$,

$$
\begin{aligned}
x^T G x &= \sum_{i=1}^{M}\sum_{j=1}^{M} \operatorname{Tr}(x_i^T \mathscr{R}(Y_{ij}) x_j) \\
&= \sum_{i=1}^{M}\sum_{j=1}^{M} \operatorname{Tr}(Y_{ij} \mathscr{R}^*(x_i x_j^T)) = \operatorname{Tr}(Y(xx^T \otimes I_3)) \geq 0
\end{aligned} \tag{3.51}
$$

if $Y \succeq 0$.

3.4.1 Convex relaxation for structural calculation: RDC-NOE-SDP and RDC-SDP

When solving (P1) in the context of protein structural calculation from RDC and NOE, we name the proposed method *RDC-NOE-SDP*. The RDC cost (3.8) in terms of Y is defined as

$$f^{\mathrm{RDC}}(Y) = \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{E_{\mathrm{RDC}i}} |\mathrm{Tr}((\nu_{nm}^{(i)}\nu_{nm}^{(i)\,T} \otimes S^{(j)})Y_{ii}) - r_{nm}^{(j)}|^p. \tag{3.52}$$

As for NOE, we simply note that the squared distances $\|\zeta_m^{(i)} - \zeta_n^{(j)}\|_2^2$ for $(m,n) \in E_{\mathrm{NOE}}$ are quadratic in R_i's (see Eq. (3.9)). Therefore the cost (3.10) can be written as

$$f^{\mathrm{NOE}}(Y) = \max((d_{mn}^{\mathrm{low}})^2 - \mathrm{Tr}(A_{mn}Y), 0)^p + \max(\mathrm{Tr}(A_{mn}Y) - (d_{mn}^{\mathrm{up}})^2, 0)^p \tag{3.53}$$

using some coefficient matrices A_{mn}'s.

Given only RDC measurement, we can solve (P2) with the RDC cost target equation (3.8) to achieve a speed-up through reduction in variable size because the cost $f^{\mathrm{RDC}}(Y)$ is of the form of equation (3.42). We call this method *RDC-SDP*.

3.4.2 Rounding: projection and manifold optimization

In this section, we detail a *rounding* scheme to extract rotations from the solutions of (P1) and (P2). We first examine the case of rounding from the solution of (P1). Denote the solution to (P1) as $Y^\star, R^\star$. When we apply the convex relaxation in (P1), it is possible that $Y^\star \neq \mathrm{vec}(R^\star)\mathrm{vec}(R^\star)^T$. To round, we first apply a rank 1 approximation to $Y^\star$ via the eigen-decomposition

$$Y^\star = \sum_i \lambda_i w_i w_i^T. \tag{3.54}$$

The rank-1 approximation to $Y^\star$ is then $y^\star y^{\star T}$, where

$$y^\star = \sqrt{\lambda_1} w_1 \tag{3.55}$$

and λ_1 and w_1 are the top eigenvalue and eigenvector of $Y^\star$. We treat $y^\star$ as a vector composed of M blocks of 9×1 smaller vectors and use $y_i^\star$ to denote the i-th block of $y^\star$. To recover individual rotations, let

$$\tilde{R}_i = \underset{R \in \mathbb{O}(3)}{\operatorname{argmin}} \|R - \operatorname{mat}(y_i^\star)\|_F^2 \tag{3.56}$$

where $\mathbb{O}(3)$ is the group of orthogonal 3×3 matrices. For any matrix A, its closest orthogonal matrix in Frobenius norm is given by by UV^T where the orthogonal matrices $U, V \in \mathbb{R}^{3\times 3}$ are obtained from the singular value decomposition (SVD) $U\Sigma V^T$ of A. Notice that $y^\star$ has a sign ambiguity and we choose the sign of $y^\star$ such that $\det(\operatorname{mat}(y_i^\star)) > 0$ (and hence $\det(\tilde{R}_i) > 0$) for the majority of $\det(\operatorname{mat}(y_i^\star))$'s. For those $\operatorname{mat}(y_i^\star)$ with negative determinants, we use

$$U \operatorname{diag}([1, 1, -1]) V^T$$

as the projection of $\operatorname{mat}(y_i^\star)$ to the nearest special orthogonal matrix after SVD (also known as Kabsch algorithm [84]). When dealing with clean data, we expect $\det(\operatorname{mat}(y_i^\star)) > 0$ for all i with the proper choice of the global sign. Even in the presence of noise, $\det(\operatorname{mat}(y_i^\star))$ is rather stable and we have not encountered a case where $\det(\operatorname{mat}(y_i^\star))$ turns out to be negative in our numerical simulation study.

A similar rounding procedure can be applied after using (P2). After obtaining the rank-1 approximation $y_i^\star y_i^{\star T}$ to $Y^{(i)\star}$, we find $\tilde{R}_i$ from

$$\tilde{R}_i = \underset{R \in \mathbb{O}(3)}{\operatorname{argmin}} \|R - \det(\operatorname{mat}(y_i^\star))\operatorname{mat}(y_i^\star)\|_F^2. \tag{3.57}$$

Notice that although it is possible to directly round $R_i^\star$ obtained from (P1) and (P2), empirically we observe obtaining the rotations from $y_i^\star$ is more robust to noise.

For the case when the solutions to (P1) and (P2) are not rank-1, the non-convex

problem of finding the rotations of the rigid units is not solved exactly. After rounding there is no guarantee that $\tilde{R}_i$ orient the rigid units optimally such that the costs (3.8) and (3.10) are minimized. In this case, since the pose recovery problem for an articulated structure is an optimization problem on the product of $\mathbb{SO}(3)$ manifolds, we use the manifold optimization toolbox Manopt [21] to refine $\tilde{R}_i$ further in order to obtain a solution with a lower cost. However, since Manopt only handles unconstrained optimization problem on a Riemanian manifold, we have to use the penalty method to handle the hinge constraint (3.6) of the type $h(R) = 0$ by adding a penalty $(\mu/2)\|h(R)\|_2^2$ with increasing μ. We note that without a good initialization, manifold optimization can easily get stuck in a local minima as it is essentially a gradient descent based approach that descends along the geodesics of a manifold.

3.4.3 Summary of the structural calculation algorithm

In this subsection we summarize the full procedures of RDC-NOE-SDP for structural calculation. The procedure of RDC-SDP follows similarly. We first solve the convex relaxed program (P1) to find the rotations that orient each rigid unit, under the hinge constraints that chain the rigid units together. Since the solution to (P1) does not necessarily yield transformations that satisfy the special orthogonality constraints, a rounding procedure detailed in Section 3.4.2 is employed to ensure special orthogonality. Using this approximate solution as a starting point, we further optimize the cost in (P1) locally using the Manopt toolbox. The estimated rotations are then used to construct the backbone coordinates using the recursive relation introduced in (3.4), and we denote these coordinates as $\zeta_k^{(i)^\star}$.

Algorithm 4 RDC-NOE-SDP

Require: Local coordinates $x_k^{(i)},\ k=1,\dots,K,\ i=1,\dots,M$, RDC and Saupe tensors in N alignment media, and NOE measurements.

Ensure: Global coordinates $\zeta_k^{(i)^\star},\ k=1,\dots,K,\ i=1,\dots,M$.

1: Find the solution $Y^\star$ to problem (P1) with cost (3.52) and (3.53) using CVX.

2: Compute the top eigenvector $y^\star$ of $Y^\star$.

3: For $i \in [1,M]$, $\tilde{R}_i = \underset{R\in\mathbb{O}(3)}{\operatorname{argmin}} \|R - \operatorname{mat}(y_i^\star)\|_F^2$. Pick the sign of $y^\star$ such that $\det(\operatorname{mat}(y_i^\star)) > 0$ for most $\operatorname{mat}(y_i^\star)$. Use Kabsch algorithm to project $\operatorname{mat}(y_i^\star)$ to $\mathbb{SO}(3)$ if $\det(\operatorname{mat}(y_i^\star)) < 0$.

4: Refine $\tilde{R}_i,\ i=1,\dots,M$ locally (e.g., using Manopt).

5: $\zeta_k^{(1)^\star} = \tilde{R}_1(x_k^{(1)} - x_{J_1}^{(1)})$, $\zeta_k^{(i)^\star} = \tilde{R}_i(x_k^{(i)} - x_{J_i}^{(i)}) + \zeta_{J_i}^{(i-1)^\star}$ for $i \in [2,M]$.

3.5 Estimating pairwise translations between multiple protein fragments

In the presence of RDC measurements, the backbone conformation of the full protein can be determined from the calculated R_i's, up to a global translation. However, it is usually the case that some of the amino-acid residues contain very few or no RDC's being measured. While RDC-SDP will certainly fail in these situations, using RDC-NOE-SDP is also undesirable. As mentioned in Section 3.4, when the NOE set is the main constraint placed on the protein structure, it is unnecessary to use (P1) but instead, a smaller convex relaxation (3.50) can be used. The convex relaxation in (P1) is typically not tight if the geometric constraints mainly come from the NOE data. In this case we need to break up the protein and calculate the conformations for selected fragments of the protein backbone. Therefore it is necessary to figure out the relative translation between the fragments in order to combine the backbone segments coherently. In this section, we propose a semidefinite relaxation that *jointly* uses NOE restraints between all fragments to piece them together. Let there be F fragments. We denote the coordinate of the k-th atom in the i-th fragment as $z_k^{(i)}$. We note that in this section, the superscript "(i)" is no longer used as the index for rigid peptide plane or CA-body, but as the index of a fragment composed of multiple amino acid residues. The goal is to find $t_1,\dots,t_F \in \mathbb{R}^3$

such that

$$(d_{kl}^{\text{low}})^2 \leq \|z_k^{(i)} + t_i - (z_l^{(j)} + t_j)\|_2^2 \leq (d_{kl}^{\text{up}})^2, \quad (k,l) \in E_{\text{NOE}} \tag{3.58}$$

It should be understood that in this context, E_{NOE} only contains the NOE distance restraints between the fragments. The squaring of the constraint is important to obtain a semidefinite relaxation to solve for the pairwise translations. Now let

$$T = \begin{bmatrix} t_1^T \\ \vdots \\ t_N^T \\ I_3 \end{bmatrix} \begin{bmatrix} t_1 & \cdots & t_N & I_3 \end{bmatrix} = \begin{bmatrix} t_1^T t_1^T & \dots & t_1^T t_F & t_1^T \\ \vdots & \ddots & \vdots & \vdots \\ t_F^T t_1 & \dots & t_F^T t_F & t_F^T \\ t_1 & \dots & t_F & I_3 \end{bmatrix} \in \mathbb{R}^{(3+F)\times(3+F)} \tag{3.59}$$

where T is rank 3 and positive semidefinite. Again, by writing (3.58) in terms of T and by relaxing the rank 3 constraint for T we can solve for the pairwise translations through the following semidefinite program

$$\text{(P3)} \quad \min_{\substack{T\succeq 0,\\ e_{kl}^{\text{up}}\geq 0,\ e_{kl}^{\text{low}}\geq 0}} \sum_{(k,l)\in E_{\text{NOE}}} e_{kl}^{\text{up}} + e_{kl}^{\text{low}} - \gamma \text{Tr}(T) \tag{3.60}$$

$$\begin{aligned} \text{s.t.} \quad & 2(T(F+1:F+3,i) - T(F+1:F+3,j))^T (z_k^{(i)} - z_l^{(j)}) \\ & + T(i,i) + T(j,j) - 2T(i,j) + \|z_k^{(i)} - z_l^{(j)}\|_2^2 \leq (d_{kl}^{\text{up}})^2 + e_{kl}^{\text{up}}, \quad (k,l) \in E_{\text{up}}, \\ & 2(T(F+1:F+3,i) - T(F+1:F+3,j))^T (z_k^{(i)} - z_l^{(j)}) \\ & + T(i,i) + T(j,j) - 2T(i,j) + \|z_k^{(i)} - z_l^{(j)}\|_2^2 \geq (d_{kl}^{\text{low}})^2 - e_{kl}^{\text{low}}, \quad (k,l) \in E_{\text{low}}, \\ & T(F+1:F+3, F+1:F+3) = I_3 \\ & T\mathbf{1} = 0. \end{aligned} \tag{3.61}$$

The last constraint is simply to remove the global translation ambiguity. Instead of using (3.58) as hard constraints to find pairwise translations that satisfy them, we penalize the violation of such bounds through the cost in (P3). This is necessary

because errors in estimating individual fragment coordinates and also ambiguous NOE assignments may cause violations of (3.58). After obtaining the solution $T^\star$, we simply use $T^\star(F+1:F+3, 1:F)$ as the translations for the fragments. The extra maximum variance unfolding [155] type regularization $-\gamma \mathrm{Tr}(T)$ prevents the fragments from clustering too tightly by maximizing the spread of the translations [17].

We conclude this section with a toy example that demonstrates the superiority of joint translation estimation using SDP. For the convenience of illustration, we provide the example in 2D. In order to sequentially assemble the fragments from pairwise distances, it is necessary that there is a pair of fragments where there are at least two distance measurements between them. This is needed to fix the relative translation between the two fragments with two degrees of freedom. In the toy example in Figure 3.2, this necessary condition for greedy sequential methods is not satisfied, but even so with (P3) we are able to recover the correct positions of the fragments. This property of (P3) is quite important, since in practice there are typically only a few NOE restraints between secondary elements of the protein backbone (with the exception of β strands) [109].

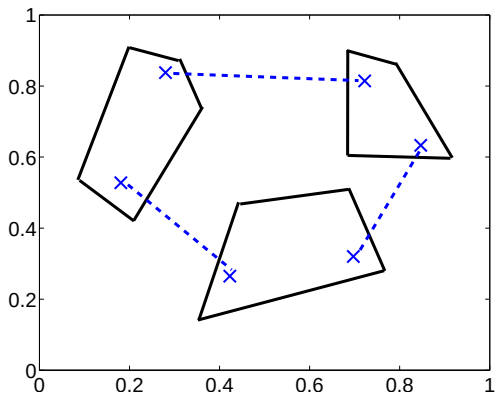


Figure 3.2: Three fragments in 2D positioned by (P3) using the distance measurements (Blue dotted lines). While it is impossible to determine the translations sequentially with the distance measurement pattern shown here, with (P3) the three fragments can be assembled jointly.

3.6 Numerical experiments

3.6.1 Comparison to Cramér-Rao lower bound

In this section, we present the results of numerical simulations with synthetic data for RDC-SDP and RDC-NOE-SDP. We first describe the noise model in our simulations. Let $\zeta = [\zeta_1, \ldots, \zeta_K] \in \mathbb{R}^{3\times K}$ be the ground truth coordinates. We drop the superscript "(i)" when denoting the atom coordinate since the membership of an atom to a rigid unit is immaterial here. Now let E_{RDC} be the set of atom pairs with RDC measured, and assume that the RDC measurements are generated through

$$r_{nm}^{(j)} = {v_{nm}}^T S^{(j)} v_{nm} + \sigma \epsilon_{nm}^{(j)}, \quad (n,m) \in E_{\text{RDC}},\ j = 1,2, \tag{3.62}$$

where the bond direction v_{nm} is related to the coordinates ζ_n, ζ_m through

$$v_{nm} = \frac{\zeta_n - \zeta_m}{\|\zeta_n - \zeta_m\|_2}. \tag{3.63}$$

We assume $\epsilon_{nm}^{(j)} \sim \mathcal{N}(0,1)$ where $\mathcal{N}(0,1)$ is the standard normal distribution. While it is quite common for different types of atomic pairs with RDC measured at different levels of uncertainty, in this section we assume r_{nm}'s are all corrupted by i.i.d. Gaussian noise of same variance σ^2 for the noise model introduced in (3.62).

In this simulation study, we use the α helix of the protein ubiquitin (residue 24 - residue 33) to generate synthetic RDC data. The data file for the PDB entry 1D3Z contains RDC datasets measured in two alignment media. From the known PDB structure, we determine the two Saupe tensors $S^{(1)}, S^{(2)}$ in these alignment media and use them for simulation purposes. We simulate synthetic RDC data using the noise model (3.62) where atom pair directions are obtained from the ground truth PDB model. For this simulation we use the pairs $\{(\text{C},\text{CA}),(\text{C},\text{N}),(\text{N},\text{H})\}$ from the peptide plane, and $\{(\text{CA},\text{HA}),(\text{CA},\text{CB})\}$ from the CA-body to generate RDCs, as the RDCs associated with

these pairs are commonly measured. In addition to RDC measurements, we also run the simulation with the aid of 16 NOE restraints on the backbone. The form of NOE restraints is in terms of upper and lower bounds. To measure the quality of a coordinate estimator $\hat{\zeta}$, we use the Root-Mean-Square-Distance (RMSD)

$$\text{RMSD} = \sqrt{\frac{\|\hat{\zeta} - \zeta\|_F^2}{K}} \tag{3.64}$$

where ζ is the starting PDB model. We evaluate the RMSD for the atoms CA, CB, C, N, H, O and HA in all amino acids.

We present the simulation results in Figure 3.3. We simulate RDC noise with $\sigma \in [0, 5e-5]$. Every data point is averaged over 30 noise realizations of RDC. We compare the scenarios of running (1) RDC-SDP without the chirality constraint (3.46), (2) RDC-SDP and (3) RDC-NOE-SDP with hard distance constraints provided by NOE, both with and without Manopt refinement after the $\mathbb{SO}(3)$ projection step. When there is no noise, for all scenarios RDC-SDP and RDC-NOE-SDP exactly recover the rotations. This is a property that simulated annealing based methods do not enjoy, as even without noise these methods can still suffer from local minima issue. The simulation also highlights the importance of the unit chirality constraint (3.46), as without such constraint RDC-SDP fails to attain 1 Å RMSD at high noise level. If the chirality constraint is included, we can achieve within 1 Å RMSD even without Manopt refinement. As expected, the inclusion of NOE measurements in RDC-NOE-SDP can further reduce the RMSD. We also compare the results of various schemes both before and after Manopt refinement, in order to show that local refinement has limited effect on the solution quality hence it is crucial to have a high quality initialization. We further compare our results against the Cramér-Rao lower bound. The CRB provides an information-theoretic lower bound for the least possible variance that can be achieved by any coordinate estimator. The derivation of the CRB is given in Section 3.8.2. With RDC-SDP we are able to attain the CRB for moderate noise

levels. In the case of RDC-NOE-SDP the CRB is attained at all noise levels. Here we remark that we slightly abused terminology by referring to the normalized RDC as RDC, where the un-normalized RDC is defined in (1.7). We emphasize that when $\sigma = 5\text{e-}5$, the magnitude of noise on the un-normalized RDC is rather large. For example, since the dipolar coupling constant for the N-H RDC is about 23 kHz, when $\sigma = 5\text{e-}5$ the actual noise is 1.15 Hz. This is larger than the typical experimental uncertainty of N-H RDC (<0.5 Hz) [78].

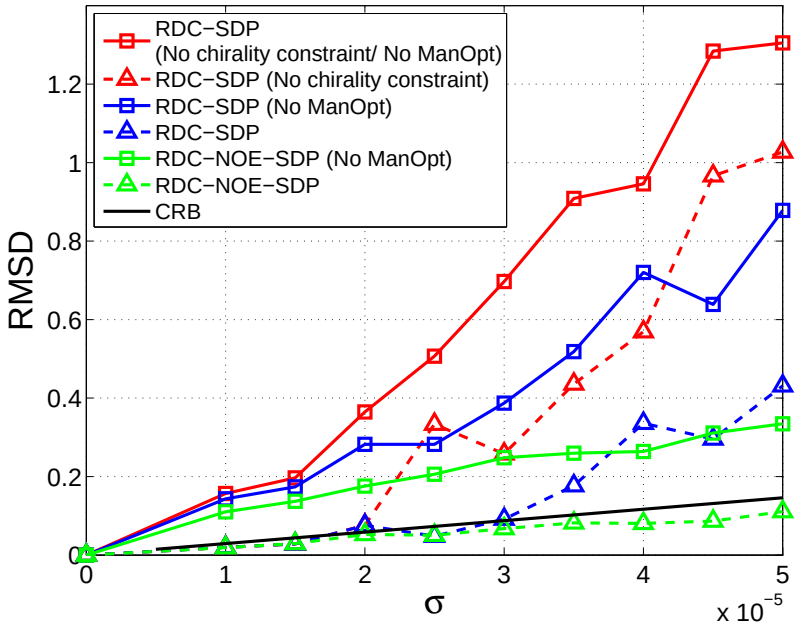


Figure 3.3: Comparison between running RDC-SDP and RDC-NOE-SDP under 6 different scenarios. We run RDC-SDP with and without the chirality constraints (3.46) both before and after Manopt refinement. When we include NOE restraints using RDC-NOE-SDP, the results are significantly improved and we are able to attain the CRB after Manopt refinement.

We also provide a comparison of our methods with the molecular fragment replacement (MFR) method proposed in [11] using the full ubiquitin sequence with 76 amino acids and about 500 backbone atoms. We first give a brief introduction to the MFR method. MFR is an RDC-based method that determines the structure of a protein through finding homologous structures in the PDB for short fragments of the protein. For a short fragment, candidate structures from the PDB are used to construct the coordinates in

(3.1). Then a least-squares procedure detailed in the introduction is used to obtain the Saupe tensor based on the experimentally measured RDC and the candidate structure. If a PDB candidate structure gives a low residual in the least-squares fitting, it will be deemed a structure similar to the protein fragment under inspection. Other experimental information such as chemical shifts can also be compared to the information recorded in the database to find a similar structure. The homologous structures for short fragments of the protein are then merged and simulated annealing is applied to further refine the structure based on the RDC measurements. In this numerical study, we start simulated annealing with temperature of 600 K and cool down to 0 K in 30000 steps. For a fair comparison between MFR and our proposed methods, we do not use chemical shift information for the MFR procedure but only RDC and NOE. We again simulate RDC measurements from the noise model in (3.62) for the bonds $(\mathrm{C},\mathrm{CA}),(\mathrm{C},\mathrm{N}),(\mathrm{N},\mathrm{H}),(\mathrm{CA},\mathrm{HA}),(\mathrm{CA},\mathrm{CB})$, with noise levels $\sigma = 2.5, 5e-5$. We supplement the RDC with 187 experimentally reported backbone NOE's. For ubiquitin, the experimentally measured backbone NOE restraints have very few violations. The RMSD of five reconstructed ubiquitin fragments, each having 13 amino-acids on average, is reported in Table 3.1. Here we use the same fragments as in Section 3.6.2 where experimental data is used to reconstruct the ubiquitin structure. The choice of the fragments will be detailed in Section 3.6.2 and Table 3.2. The overall RMSD of the protein backbone is also reported after assembling the five fragments using (P3) in the last column of Table 3.1. It is shown that the total RMSD obtained from RDC-SDP and RDC-NOE-SDP is significantly lower than the RMSD from the MFR method. Since MFR relies heavily on initialization, when the noise is high, the identification of a wrong homologous structure can severely impact the solution quality of simulated annealing. It is expected that RDC-NOE-SDP outperforms RDC-SDP, at the expense of using more data, as in Figure 3.3. The total RMSD of the entire backbone is generally higher than the RMSD of the fragments, due to the imprecision of the NOE restraints and error

accumulation when assembling the fragments.

		1	2	3	4	5	Full backbone
σ=2.5e-5	RDC-SDP	0.27	0.49	0.71	0.19	0.47	1.75
	RDC-NOE-SDP	0.27	0.32	0.10	0.11	0.53	1.35
	MFR	1.13	1.78	0.96	1.77	2.74	2.87
σ=5e-5	RDC-SDP	0.67	0.59	1.07	0.86	2.34	2.34
	RDC-NOE-SDP	0.40	0.48	0.27	0.57	0.84	1.72
	MFR	1.44	2.22	1.24	2.92	2.83	4.43

Table 3.1: RMSD (Å) of five ubiquitin fragments using RDC-SDP, RDC-NOE-SDP and MFR from simulated data with noise levels σ=2.5e-5 and 5e-5. The residue number in each fragment is reported in Table 3.2. The results are averaged over 10 noise realizations.

3.6.2 Experimental data for ubiquitin

In this section, we present results on the analysis of experimental RDC data obtained in two alignment media for ubiquitin. We only consider the peptide planes and CA-bodies coming from the first 70 amino acids since the last 6 residues are highly flexible and do not contribute to rigid constraints. In real data there are on average 7 RDC measurements per amino acid in two different alignment media, arising from the bonds $(C, CA), (C, N)$, $(N, H), (CA, HA), (CA, CB)$. Unlike the simulated case, in experimental data there might be missing RDC measurements for some bonds. We again supplement the RDC with 187 experimentally reported backbone NOE's. We use both RDC-SDP and RDC-NOE-SDP to solve the backbone structure of five ubiquitin fragments, each containing 12-13 residues on average. We split the fragments at amino-acid sites where there are too few or no RDC measurements. The results are summarized in Table 3.2. When using only RDC, it is more difficult to determine the backbone structure near the starting and end point of a fragment since RDC measurements are generally sparser in those regions. Therefore the fragments we used for RDC-SDP sometimes have smaller size than the fragments used for RDC-NOE-SDP which uses additional distance measurements. Typically, when

using RDC-SDP, we can tell whether the rotation for a rigid unit is well-determined by simply examining how well $Y^{(i)\star}$ can be approximated by a rank 1 matrix. We can exclude those rigid units near the end of a fragment that give rise to high rank solutions when solving (P2). In terms of accuracy, due to the additional distance restraints, RDC-NOE-SDP outperforms RDC-SDP. The average RMSD of the fragments are 0.67 Å and 0.57 Å for RDC-SDP and RDC-NOE-SDP respectively when comparing with the X-ray structure 1UBQ. To provide a different perspective, we also compare the results from our method with the high resolution NMR structure 1D3Z [38]. Since RDC-SDP only involves PSD variables of size 9×9, whereas RDC-NOE-SDP involves variable of size $9M \times 9M$, the running time of RDC-SDP is significantly lower than RDC-NOE-SDP. In particular, the running time for (P2) in RDC-SDP is never more than 2 seconds but the running time for (P1) in RDC-NOE-SDP can be as long as 5 minutes. When we combine the fragments using (P3), the conformation errors of the whole protein backbone obtained from fragments determined by RDC-SDP and RDC-NOE-SDP are 1.28 (1.25) Å and 1.07 (1.11) Å RMSD respectively when comparing to 1UBQ (1D3Z). In practice when calculating the protein backbone structure, we may want to use RDC-SDP instead of RDC-NOE-SDP to obtain an initial structure and add NOE measurements in the local refinement stage if running time is a concern. Figure 3.4 further compares the backbone traces obtained from our proposed methods and the X-ray structure. We also compare our results with MFR in Table 3.2. Comparing to RDC-SDP or RDC-NOE-SDP, structures calculated from MFR has a closer similarity to the X-ray structure 1UBQ, with average fragment RMSD and overall RMSD being 0.54 Å and 0.87 Å respectively. Since our proposed methods have not yet taken into accounts potential terms concerning radius of gyration, Van der Waals lower bound and infeasibility of the torsion angles, it is reasonable that the proposed methods still cannot compare with MFR.

		1	2	3	4	5
Residue No.	RDC-SDP	2-7	10-18	22-36	39-53	54-70
	RDC-NOE-SDP	1-7	10-18	22-36	37-53	54-70
	MFR	2-7	10-18	22-36	39-53	54-70
RMSD (Å)	RDC-SDP	0.57	0.51	0.81	0.70	0.78
1UBQ	RDC-NOE-SDP	0.41	0.54	0.71	0.54	0.65
	MFR	0.42	0.51	0.45	0.78	0.52
RMSD (Å)	RDC-SDP	0.56	0.48	0.78	0.62	0.73
1D3Z	RDC-NOE-SDP	0.42	0.52	0.72	0.47	0.59
	MFR	0.40	0.46	0.42	0.71	0.44
Time (s)	RDC-SDP	8 (0.5)	11 (0.5)	63 (2)	22 (1)	23 (1.3)
	RDC-NOE-SDP	15 (6)	30 (17)	231 (162)	596 (450)	312 (281)
	MFR	1560 (all 5 fragments)				

Table 3.2: Results of computing the structure of five ubiquitin fragments using RDC-SDP, RDC-NOE-SDP and MFR from experimental data. We compare with both the X-ray structure 1UBQ and the high resolution NMR structure 1D3Z. The time in brackets is the running time of the SDPs (P1) and (P2) used by RDC-NOE-SDP and RDC-SDP. The excess time is due to Manopt refinement. For MFR we only report the total running time for calculating the entire backbone.

3.7 Conclusion

We present two novel convex relaxations RDC-SDP and RDC-NOE-SDP to calculate the protein backbone conformation from both RDC and NOE measurements. In simulations, our methods exactly recover the protein structure when there is no noise, whereas simulated annealing based methods can still suffer from local minima issue even when the data is clean. In the presence of noise, the error of our solution comes close to the CRB. We illustrate the robustness of our methods through comparing with the popular MFR homology modelling method in the high-noise regime in simulations. We further demonstrated the success of our methods by obtaining a backbone structure of 1 Å

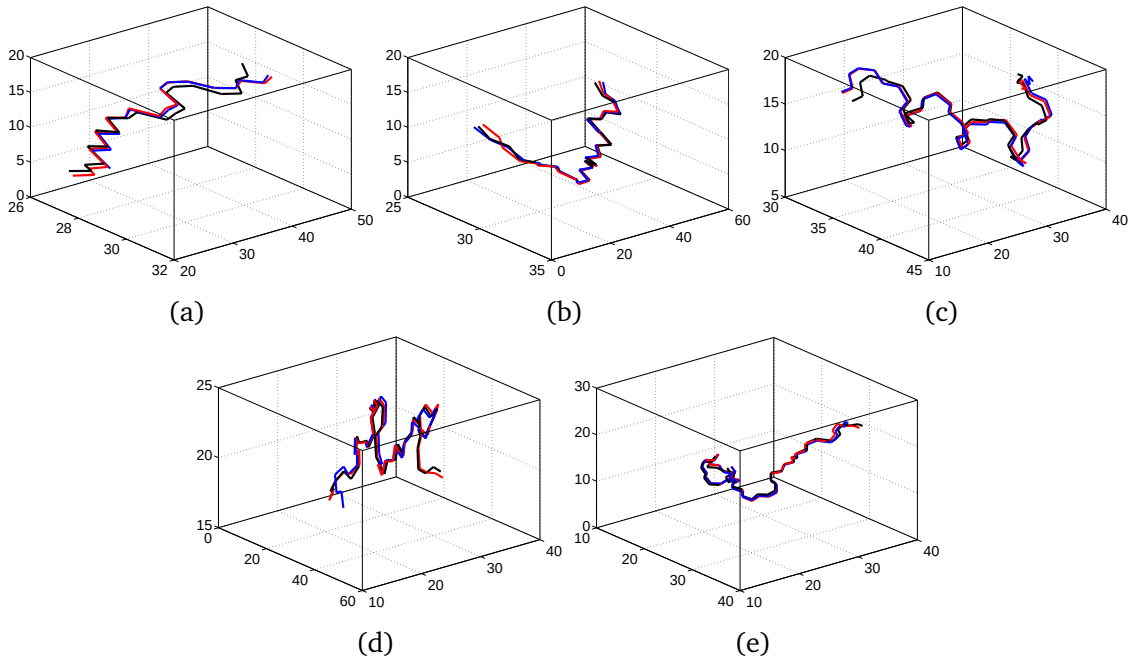


Figure 3.4: The trace of protein backbone drawn using N, CA and C. The black, blue and red curves come from the X-ray model 1UBQ, RDC-SDP solution and RDC-NOE-SDP respectively.

resolution for ubiquitin using real experimental data. Both proposed methods are fast in practice, in particular RDC-SDP can determine a protein fragment of typical size in just a few seconds. This is in sharp contrast to current methods such as MFR, RDC-Analytics and REDCRAFT that have running time ranging from tens of minutes to two hours. This property of our algorithm can be useful when iterating between estimating resonance or NOE assignments and structural calculation [67]. In a broader context, the proposed methods can also be applied to pose estimation problems for articulated structure in computer vision and robotics.

3.8 Appendix

3.8.1 Unit quaternions and quadratic problem on $\mathbb{SO}(3)$

We first give a brief introduction to unit quaternions, where a detailed exposition can be found in many other sources (e.g. [5]). The group of unit quaternions consists of elements of the form

$$q = a + b\mathbf{i} + c\mathbf{j} + d\mathbf{k}$$

which is a linear combination of the basis $1, \mathbf{i}, \mathbf{j}, \mathbf{k}$ and

$$a^2 + b^2 + c^2 + d^2 = 1$$

The basis satisfies the multiplication rules

$$\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -1 \tag{3.65}$$

and these define the multiplication of any two quaternions. It is easy to see that the inverse q^{-1} of a quaternion q is

$$q^{-1} = a - b\mathbf{i} - c\mathbf{j} - d\mathbf{k}$$

The group of unit quaternions can be used to represent a rotation in $\mathbb{SO}(3)$. If we parameterize the unit quaternion as $q = \cos(\theta/2) + \sin(\theta/2)(u_x\mathbf{i} + u_y\mathbf{j} + u_z\mathbf{k})$ it can be regarded as a rotation around the axis $[u_x, u_y, u_z]^T \in \mathbb{R}^3$ by angle θ. More precisely, if we are to rotate any vector $v \in \mathbb{R}^3$ using a quaternion, we simply let

$$\tilde{v} = 0 + v(1)\mathbf{i} + v(2)\mathbf{j} + v(3)\mathbf{k}$$

and the rotation on v is applied through

$$q\tilde{v}q^{-1} = 0 + a\mathbf{i} + b\mathbf{j} + c\mathbf{k} \tag{3.66}$$

The coefficients in front of $\mathbf{i}, \mathbf{j}, \mathbf{k}$ give the rotated v in $\mathbb{R}^3$. Notice that q and $-q$ result in the same rotation on the vector v. From (3.66), a relation between rotation matrices in $\mathbb{SO}(3)$ and unit quaternions can be obtained (also known as Euler-Rodrigues formula). If we treat the unit quaternion q as a vector in $\mathbb{R}^4$ such that $\|q\|_2 = 1$, the rotation matrix it represents is given by

$$\phi(q) = \begin{bmatrix} 1-2q(3)^2-2q(4)^2 & 2(q(2)q(3)-q(4)q(1)) & 2(q(2)q(4)+q(3)q(1)) \\ 2(q(2)q(3)+q(4)q(1)) & 1-2q(2)^2-2q(4)^2 & 2(q(3)q(4)-q(2)q(1)) \\ 2(q(2)q(4)-q(3)q(1)) & 2(q(3)q(4)+q(2)q(1)) & 1-2q(2)^2-2q(3)^2 \end{bmatrix}. \tag{3.67}$$

This map ϕ is a surjective group homomorphism (epimorphism) from the group of unit quaternions to $\mathbb{SO}(3)$. The kernel of this map is $\{[-1,0,0,0]^T, [1,0,0,0]^T\}$. This implies for a matrix $R \in \mathbb{SO}(3)$, $R = \phi(q) = \phi(-q)$ for a quaternion q. Therefore the group of unit quaternions is known as the *double cover* of $\mathbb{SO}(3)$, in other words,

$$\{\{q, -q\} \mid q \in \mathbb{R}^4, \|q\|_2 = 1\} \cong \mathbb{SO}(3). \tag{3.68}$$

In light of this, if we construct the following set of rank-1 matrices

$$\text{Quaternion}^2 := \{Q \in \mathbb{R}^{4\times 4} \mid Q = qq^T, \|q\|_2 = 1\} \tag{3.69}$$

and define a function Φ via ϕ as

$$\Phi(qq^T) := \phi(q), \tag{3.70}$$

then the map

$$\Phi : \text{Quaternion}^2 \to \mathbb{SO}(3) \tag{3.71}$$

is a bijection. It can be checked easily that the inverse map Φ^{-1} is

$$\Phi^{-1}(R) := \frac{1}{4}\begin{bmatrix} 1+R(1,1)+R(2,2)+R(3,3) & R(3,2)-R(2,3) & R(1,3)-R(3,1) & R(2,1)-R(1,2) \\ R(3,2)-R(2,3) & 1-R(2,2)-R(3,3)+R(1,1) & R(1,2)+R(2,1) & R(1,3)+R(3,1) \\ R(1,3)-R(3,1) & R(1,2)+R(2,1) & 1-R(3,3)-R(1,1)+R(2,2) & R(2,3)+R(3,2) \\ R(2,1)-R(1,2) & R(1,3)+R(3,1) & R(2,3)+R(3,2) & 1-R(1,1)-R(2,2)+R(3,3) \end{bmatrix} \tag{3.72}$$

The bijection between Quaternion^2 and $\mathbb{SO}(3)$ leads to the simple proposition.

Proposition 3.8.1. *$R \in \mathbb{SO}(3)$ if and only if $\Phi^{-1}(R) \in$ Quaternion^2.*

Proposition 3.8.1 shows that we can use the constraint $\Phi^{-1}(R) \in \text{Quaternion}^2$ to enforce $R \in \mathbb{SO}(3)$. Notice that

$$\Phi^{-1}(R) \in \text{Quaternion}^2 \tag{3.73}$$

implies

$$\Phi^{-1}(R) = qq^T \text{ for some } q \in \mathbb{R}^4,\ \|q\|_2 = 1 \tag{3.74}$$

hence

$$\Phi^{-1}(R)\Phi^{-1}(R) = \Phi^{-1}(R) \tag{3.75}$$

This gives a linear constraint in Y and R. Indeed, if

$$\text{vec}(\Phi^{-1}(R)) := A \begin{bmatrix} \text{vec}(R) \\ 1 \end{bmatrix}$$

for some matrix $A \in \mathbb{R}^{16\times 10}$, then

$$\Phi^{-1}(R)\Phi^{-1}(R) \;=\; \sum_{i=1}^{4}\left(A \begin{bmatrix} \text{vec}(R) \\ 1 \end{bmatrix}\begin{bmatrix} \text{vec}(R)^T & 1 \end{bmatrix}A^T\right)_{ii}$$

$$
\begin{aligned}
&= \sum_{i=1}^{4} (A \begin{bmatrix} Y & \text{vec}(R) \\ \text{vec}(R)^T & 1 \end{bmatrix} A^T)_{ii} \\
&= \Psi\Big(\begin{bmatrix} Y & \text{vec}(R) \\ \text{vec}(R)^T & 1 \end{bmatrix} \Big) \qquad (3.76)
\end{aligned}
$$

where $\Psi : \mathbb{R}^{4\times4} \to \mathbb{R}^{4\times4}$ is yet another linear operator. Specifically in (3.76), for a matrix $X \in \mathbb{R}^{16\times16}$ we use X_{ii} to denotes the i-th 4×4 block on the diagonal. In this way, (3.75) can be written as

$$
\Phi^{-1}(R) = \Psi\Big(\begin{bmatrix} Y & \text{vec}(R) \\ \text{vec}(R)^T & 1 \end{bmatrix} \Big). \qquad (3.77)
$$

It can be verified that any R that satisfies the last constraint in (3.24) also satisfies (3.77). This leads to the fact that R in (3.24) belongs to the convex hull of $\mathbb{SO}(3)$. To see this, notice that if $Y \succeq \text{vec}(R)\text{vec}(R)^T$ then

$$
\begin{bmatrix} Y & \text{vec}(R) \\ \text{vec}(R)^T & 1 \end{bmatrix} \succeq 0,
$$

and so is

$$
A \begin{bmatrix} Y & \text{vec}(R) \\ \text{vec}(R)^T & 1 \end{bmatrix} A^T
$$

and its 4×4 blocks along the diagonal. Therefore

$$
\Phi^{-1}(R) = \Psi\Big(\begin{bmatrix} Y & \text{vec}(R) \\ \text{vec}(R)^T & 1 \end{bmatrix} \Big) \succeq 0,
$$

in (3.24). We now state a theorem in [123, 124]:

Theorem 3.8.2. *[123, Proposition 1].*

$$
conv(\mathbb{SO}(3)) \quad = \quad \{R \in \mathbb{R}^{3\times3} \mid \Phi^{-1}(R) \succeq 0\}. \qquad (3.78)
$$

Leveraging the theorem, we arrive at the conclusion that R in (3.24) is in the convex hull of $\mathbb{SO}(3)$.

3.8.2 Cramér-Rao lower bound

In this section, we introduce a classical tool from statistics, the Cramér-Rao bound (CRB) [29], to give perspective on the lowest possible error any unbiased estimator can achieve when estimating coordinates from noisy RDC measurements. We first describe the CRB for general point estimators. Let $\theta \in \mathbb{R}^n$ be a multidimensional parameter which is to be estimated from measurements $x \in \mathbb{R}^m$. Suppose x is generated from the distribution $p(x|\theta)$. The Fisher information matrix is defined as the $n \times n$ matrix

$$I(\theta) = \mathbb{E}[(\nabla_\theta \ln p(x|\theta))(\nabla_\theta \ln p(x|\theta))^T] \tag{3.79}$$

where expectation is taken with respect to the distribution $p(x|\theta)$ and the gradient ∇_θ is taken with respect to θ. For any unbiased estimator $\hat{\theta}$ of θ, that is $\mathbb{E}(\hat{\theta}) = \theta$, the following relationship holds:

$$\mathbb{E}[(\hat{\theta} - \theta)(\hat{\theta} - \theta)^T] \succeq I(\theta)^{-1} \tag{3.80}$$

if $I(\theta)$ is invertible. Therefore the total variance of the estimator $\hat{\theta}$ is lower bounded by $\mathrm{Tr}(I(\theta)^{-1})$. We also introduce the CRB in the case when θ and $\hat{\theta}$ are constrained to be in the set $\{\theta |\ f(\theta) = 0\}$ where $f : \mathbb{R}^n \to \mathbb{R}^k$ [140]. Let $Df(\theta) \in \mathbb{R}^{k \times n}$ be the gradient matrix of f at θ with full row rank, and $Q \in \mathbb{R}^{n \times (n-k)}$ be a set of orthonormal vectors satisfying

$$Df(\theta)Q = 0$$

i.e. Q is an orthonormal basis of the null space of $Df(\theta)$. In this case, for any unbiased estimator $\hat{\theta}$ satisfying $f(\hat{\theta}) = 0$, the CRB is then

$$\mathbb{E}[(\hat{\theta} - \theta)(\hat{\theta} - \theta)^T] \succeq Q(Q^T I(\theta) Q)^{-1} Q^T \tag{3.81}$$

if $Q^T I(\theta) Q$ is invertible.

We are now ready to investigate the CRB for estimating atomic positions from RDC data. Let $\zeta = [\zeta_1, \ldots, \zeta_K] \in \mathbb{R}^{3\times K}$ be the coordinates of the atoms we want to estimate. We aim to derive a lower bound for $\mathbb{E}[\mathrm{Tr}((\hat{\zeta} - \zeta)^T(\hat{\zeta} - \zeta))]$ for any unbiased estimator $\hat{\zeta}$ of ζ. We assume that the RDC measurements are generated through the noise model in (3.62). We further assume that within each rigid unit, the distance between any pair of atoms is fixed. We therefore have a set of equality constraints

$$d_{nm}^2 = \|\zeta_n - \zeta_m\|_2^2, \quad (n, m) \in E_{\text{fixed}} \tag{3.82}$$

where E_{fixed} consists of all atom pairs within each and every rigid unit. Without loss of generality, we also consider the constraint

$$\zeta \mathbf{1} = 0 \tag{3.83}$$

which implies the points $\zeta_1, \ldots, \zeta_K$ are centered at zero. This is due to the fact that

$$\begin{aligned}
\mathrm{Tr}((\hat{\zeta} - \zeta)^T(\hat{\zeta} - \zeta)) &= \mathrm{Tr}((\hat{\zeta}_c - \zeta_c - t\mathbf{1}^T)^T(\hat{\zeta}_c - \zeta_c - t\mathbf{1}^T)) \\
&= \mathrm{Tr}((\hat{\zeta}_c - \zeta_c)^T(\hat{\zeta}_c - \zeta_c)) + (1/K)\|t\|_2^2 \\
&\quad -2\mathrm{Tr}((\hat{\zeta}_c - \zeta_c)^T t\mathbf{1}^T) \\
&= \mathrm{Tr}((\hat{\zeta}_c - \zeta_c)^T(\hat{\zeta}_c - \zeta_c)) + (1/K)\|t\|_2^2 \\
&\geq \mathrm{Tr}((\hat{\zeta}_c - \zeta_c)^T(\hat{\zeta}_c - \zeta_c))
\end{aligned} \tag{3.84}$$

where ζ_c and $\hat{\zeta}_c$ denote the zero centered coordinates and coordinate estimators, and t is the relative translation between ζ and $\hat{\zeta}$. Eq. (3.84) implies that deriving a lower bound for $\mathbb{E}[\mathrm{Tr}((\hat{\zeta}_c-\zeta_c)^T(\hat{\zeta}_c-\zeta_c))]$ is sufficient for obtaining a lower bound for $\mathbb{E}[\mathrm{Tr}((\hat{\zeta}-\zeta)^T(\hat{\zeta}-\zeta))]$. When there are atoms that are constrained to lie on the same plane, we need to add the constraint that any three vectors in the plane span a space with zero volume, i.e.

$$\det([\zeta_i-\zeta_j,\zeta_k-\zeta_l,\zeta_m-\zeta_n])=0 \tag{3.85}$$

for atoms i,j,k,l,m,n in the same plane.

To obtain the CRB for estimating ζ from RDC data generated through (3.62), we need to first derive an expression for the Fisher information matrix. From (3.62) and (3.63), the likelihood function for the coordinates is

$$p(\{r_{nm}\}_{(n,m)\in E_{\mathrm{RDC}}}|\zeta_1,\ldots,\zeta_K)= \prod_{\substack{(n,m)\in E_{\mathrm{RDC}}\\ j=1,2}}\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{\left((\zeta_n-\zeta_m)^TS^{(j)}(\zeta_n-\zeta_m)-r_{nm}^{(j)}d_{nm}^2\right)^2}{2d_{nm}^4\sigma^2}\right) \tag{3.86}$$

and the log-likelihood is (up to an additive constant)

$$\begin{aligned} l(\{r_{nm}\}_{(n,m)\in E_{\mathrm{RDC}}}|\zeta_1,\ldots,\zeta_K) &= \ln p(\{r_{nm}\}_{(n,m)\in E_{\mathrm{RDC}}}|\zeta_1,\ldots,\zeta_K) \\ &= -\sum_{\substack{(n,m)\in E_{\mathrm{RDC}}\\ j=1,2}}\frac{1}{2d_{nm}^4\sigma^2}((\zeta_n-\zeta_m)^TS^{(j)}(\zeta_n-\zeta_m)-r_{nm}^{(j)}d_{nm}^2)^2 \\ &= -\sum_{\substack{(n,m)\in E_{\mathrm{RDC}}\\ j=1,2}}\frac{1}{2d_{nm}^4\sigma^2}(e_{nm}^T\zeta^TS^{(j)}\zeta e_{nm}-r_{nm}^{(j)}d_{nm}^2)^2, \end{aligned} \tag{3.87}$$

where $e_{nm}=e_n-e_m$. The derivative of l with respect to $\mathrm{vec}(\zeta)$ is then

$$\nabla_{\mathrm{vec}(\zeta)}l=-\sum_{\substack{(n,m)\in E_{\mathrm{RDC}}\\ j=1,2}}\frac{2(e_{nm}^T\zeta^TS^{(j)}\zeta e_{nm}-r_{nm}^{(j)}d_{nm}^2)}{d_{nm}^4\sigma^2}(e_{nm}e_{nm}^T\otimes S^{(j)})\mathrm{vec}(\zeta). \tag{3.88}$$

It follows from the noise model (3.62) and the independence of $\epsilon_{nm}^{(j)}$'s that the Fisher information matrix

$$I(\zeta) = \mathbb{E}((\nabla_{\mathrm{vec}(\zeta)} l)(\nabla_{\mathrm{vec}(\zeta)} l)^T) \\ = 4 \sum_{\substack{(n,m)\in E_{\mathrm{RDC}} \\ j=1,2}} \frac{(e_{nm}e_{nm}^T \otimes S^{(j)})\mathrm{vec}(\zeta)\mathrm{vec}(\zeta)^T(e_{nm}e_{nm}^T \otimes S^{(j)})}{\sigma^2 d_{nm}^4}. \quad (3.89)$$

Having the Fisher information matrix, we now incorporate the constraints in (3.82) and (3.83) in order to obtain a bound as in (3.81). Stacking the equality constraints (3.82) into a $|E_{\mathrm{fixed}}| \times 1$ matrix, we get

$$f(\mathrm{vec}(\zeta)) := \left[e_{nm}^T \zeta^T \zeta e_{nm} - d_{nm}^2 \right]_{(n,m)\in E_{\mathrm{fixed}}} = 0 \quad (3.90)$$

The gradient matrix is thus

$$Df(\mathrm{vec}(\zeta)) = \mathrm{vec}(\zeta)^T \left[(e_{nm}e_{nm}^T \otimes I_3) \right]_{(n,m)\in E_{\mathrm{fixed}}} \quad (3.91)$$

where $Df(\mathrm{vec}(\zeta)) \in \mathbb{R}^{|E_{\mathrm{fixed}}| \times 3K}$. We note that $Df(\mathrm{vec}(\zeta))$ is known as the *rigidity matrix* [80], and the vectors in its null space indicate the direction of infinitesimal motion the atoms can take without violating (3.82). Even in the case when all pairwise distances between the atoms are known, there is still a 6-dimensional null space for $Df(\mathrm{vec}(\zeta))$, corresponding to an infinitesimal global rotation and translation to the coordinates ζ that preserves all pairwise distances. We now augment $f(\mathrm{vec}(\zeta)) = 0$ with the centering constraint $\zeta \mathbf{1} = 0$, and this augments $Df(\mathrm{vec}(\zeta))$ with three rows $\mathbf{1}^T \otimes I_3$, i.e.

$$Df(\mathrm{vec}(\zeta)) = \begin{bmatrix} \mathrm{vec}(\zeta)^T [(e_{nm}e_{nm}^T \otimes I_3)]_{(n,m)\in E_{\mathrm{fixed}}} \\ \mathbf{1}^T \otimes I_3 \end{bmatrix}. \quad (3.92)$$

The inclusion of such centering constraint eliminates the three dimensional subspace in

the kernel of the rigidity matrix that corresponds to the translational degree of freedom. Let Q be an orthonormal basis that spans the null space of $Df(\mathrm{vec}(\zeta))$. Together with (3.89) and (3.81) we obtain the desired CRB. We omit detailing the derivative for constraint (3.85) but simply note that the inclusion of such constraints eliminates the out of plane infinitesimal motion for atoms lying on rigid planar unit.

Inclusion of NOE constraints

We have so far neglected the use of NOE measurements when deriving the CRB. Unlike RDC, the NOE restraints remain more qualitative, with imprecise upper and lower bound [19] due to the r^{-6} scaling of the interaction. Therefore it is conventional to treat the backbone NOE as inequality constraints on distances. For an unbiased estimator $\hat{\theta}$ of the parameter θ where both $\hat{\theta}$ and θ lie in the set $\{\theta |\, f(\theta) < 0\}$, it is shown in [58] that the CRB is the same as the unconstrained case (3.80), since roughly speaking the CRB only depends on the local curvature of the log-likelihood function around θ. Therefore if the original coordinates and the coordinate estimators strictly satisfy the distance constraints (1.4), then the CRB is the same as in the case with only RDC.

Infinitesimal rigidity and invertibility of the Fisher information matrix

In this subsection, we study the infinitesimal rigidity [99] of the protein structure given RDC and distance measurements and how it guarantees invertibility of the Fisher information matrix. Let a framework with coordinates $\zeta \in \mathbb{R}^{3\times K}$ be constrained by

$$\begin{aligned}(\zeta_n - \zeta_m)^T(\zeta_n - \zeta_m) &= d_{nm}^2, \quad (n,m) \in E_{\text{fixed}},\\ (\zeta_n - \zeta_m)^T S^{(j)}(\zeta_n - \zeta_m) &= r_{nm}^{(j)}, \quad j = 1,\dots,N,\ (n,m) \in E_{\text{RDC}}.\end{aligned} \tag{3.93}$$

In order to derive a condition for infinitesimal rigidity, we first let $\mathrm{vec}(\zeta(s))$ be a curve in dimension $\mathbb{R}^{3K}$ parameterized by s, where $\zeta(0)$ satisfies (3.93). Taking derivative of

the constraints in (3.93) with respect to s at $s = 0$, we have

$$\begin{bmatrix} \mathrm{vec}(\zeta(0))^T [e_{nm} e_{nm}^T \otimes I_3]_{(n,m)\in E_{\text{fixed}}} \\ \mathrm{vec}(\zeta(0))^T [e_{nm} e_{nm}^T \otimes S^{(j)}]_{(n,m)\in E_{\text{RDC}}, j\in[1,N]} \end{bmatrix} \frac{d}{ds}\mathrm{vec}(\zeta(0)) = R(\zeta(0))\frac{d}{ds}\mathrm{vec}(\zeta(0)) = 0. \tag{3.94}$$

The null space of the generalized rigidity matrix $R(\zeta(0))$ with dimension $(|E_{\text{fixed}}| + |E_{\text{RDC}}|) \times 3K$ represents the direction of infinitesimal motion such that $\zeta(s)$ satisfies the constraints (3.93) for infinitesimally small s. If $R(\zeta(0))$ only has a three dimensional nullspace, i.e. the global translations in x, y, z-directions, we say the framework $\zeta(0)$ along with the constraints (3.93) is infinitesimally rigid.

Now we verify that the constrained Fisher information matrix is invertible if $R(\zeta(0))$ has a three dimensional null space corresponds to global translation of the points. Let Q again be the basis of the nullspace of $Df(\mathrm{vec}(\zeta))$ defined in (3.92) such that $Df(\mathrm{vec}(\zeta))Q = 0$. Let v satisfies

$$Q^T I(\zeta) Q v = 0$$

$Q^T I(\zeta) Q v = 0$ if and only if $v \in \ker(Q)$ or $Qv \in \ker(I)$. Since the columns of Q are linearly independent, $Qv \neq 0$ unless $v = 0$. This means $Q^T I(\zeta) Q v = 0$ if and only if $v = 0$ or $Qv \in \ker(I) \cap \mathrm{range}(Q) = \ker(I) \cap \ker(Df(\mathrm{vec}(\zeta)))$. Therefore if

$$\ker(I) \cap \ker(Df(\mathrm{vec}(\zeta))) = \emptyset,$$

or in other words

$$\mathrm{span}\big(\mathrm{range}(I) \cup Df(\mathrm{vec}(\zeta))\big) = \mathbb{R}^{3K} \tag{3.95}$$

then $Q^T I(\zeta) Q$ is invertible. From the form of the (3.89), it is easy to show that the range

condition (3.95) is satisfied if and only if the range of

$$\begin{bmatrix} \mathbf{1}^T \otimes I_3 \\ \mathrm{vec}(\zeta(0))^T [e_{nm} e_{nm}^T \otimes I_3]_{(n,m)\in E_{\text{fixed}}} \\ \mathrm{vec}(\zeta(0))^T [e_{nm} e_{nm}^T \otimes S^{(j)}]_{(n,m)\in E_{\text{RDC}}, j\in[1,N]} \end{bmatrix} = \begin{bmatrix} \mathbf{1}^T \otimes I_3 \\ R(\zeta(0)) \end{bmatrix} \tag{3.96}$$

is $\mathbb{R}^{3K}$. Then we arrive at the conclusion that if the framework ζ is infinitesimally rigid with the null space of $R(\zeta)$ being the global translations, the constrained Fisher information matrix defined as $Q^T I(\zeta) Q$ is invertible.

In [166], it is shown that if there exists RDC measurements for a bond in the peptide plane and a bond in the CA-body in a single alignment media, the solutions of the protein structure form a discrete set. Therefore under this condition, there is no infinitesimal motion other than global translation such that the protein framework satisfies the RDC and NOE constraints. We can thus compute the CRB safely under such condition.

Chapter 4

Bias correction in Saupe tensor estimation

In this chapter, we study the bias in OLS Saupe tensor estimator from RDC using a template protein structure. We first show how this problem naturally arises when using RDC to enhance the global registration results. The RDC between nuclei n and m is defined through the equation

$$r_{nm} = v_{nm}^T S v_{nm} \tag{4.1}$$

where v_{nm} is the unit vector between nuclei n and m and S is the Saupe tensor. When given multiple rigid protein fragments in different coordinate systems, the fragment coordinate $x_k^{(i)}$ for the k-th atom in the i-th protein fragment is related to the global coordinate x_k via

$$x_k = O_i x_k^{(i)} + t_i, \tag{4.2}$$

where O_i is an orthogonal transform and t_i is a translation. Then

$$v_{nm} = O_i v_{nm}^{(i)} \tag{4.3}$$

where $v_{nm}^{(i)}$ is the unit vector between atom n and atom m in the coordinate system of the i-th protein fragment.

The Saupe tensor for fragment i, denoted $S^{(i)}$, can be obtained by solving the linear systems

$$r_{nm} = {v_{nm}^{(i)}}^T S^{(i)} v_{nm}^{(i)} \tag{4.4}$$

if the fragment contains at least five bonds with RDC measured. The relationship between v_{nm} and $v_{nm}^{(i)}$ in (4.3) implies that

$$O_i S^{(i)} O_i^T = S \tag{4.5}$$

and

$$S^{(i)} O_i^T O_j = O_i^T O_j S^{(j)} \tag{4.6}$$

for any $i, j = 1, \ldots, M$. Intuitively, this means the orthogonal transformation O_i that aligns the fragments must also align the Saupe tensor. The relationship in (4.6) can be directly used to enhance the global registration results using RDC by augmenting the cost function in `GRET` with a term

$$\sum_{i=1}^{M} \sum_{j>i} \| S^{(i)} G_{ij} - G_{ij} S^{(j)} \|_F^2. \tag{4.7}$$

When the protein fragments structure are not determined exactly, the estimated fragment Saupe tensor $S^{(i)}$'s are corrupted. Observe that (4.5) implies in principle, all $S^{(i)}$'s share common eigenvalues. This observation can therefore be used to denoise $S^{(i)}$'s, by simply averaging the the eigenvalues of $S^{(i)}$'s. However, the averaging procedure cannot remove the bias error of $S^{(i)}$ due to noise on bond direction $v_{nm}^{(i)}$. Therefore we study the bias error in Saupe tensor estimation and how it can be removed in the subsequent sections in this chapter.

In the remaining sections, we first illustrate how the structural noise on the bond

vectors of the template structure leads to bias in the OLS estimation of Saupe tensor parameters. By structural noise, we mean the template structure used for Saupe tensor fitting differs from the true structure of the molecule due to the flexibility of the protein. Our simulation shows that when noise is added to the torsion angles of the protein backbone, the magnitude of the estimated Saupe tensor eigenvalues are typically smaller than their ground truth value, as demonstrated in Fig. 4.1. Our observation corroborates with the simulation results reported in [173], in which i.i.d. noise is added to each bond vector instead. In linear regression, such decrease in magnitude of the estimator in the presence of noise on the regressor is commonly known as *attenuation* [28]. While the focus of [173] is mainly to use Monte Carlo simulation to evaluate the uncertainty of estimated alignment magnitude and rhombicity, we focus on using it to correct the attenuation effect in the OLS Saupe tensor eigenvalues estimator. The method we propose bears similarity with the statistical method *simulation extrapolation* (SIMEX) [37, 139] that is frequently used to correct for the attenuation effect. Typically this type of methods are parametric and require noise variance as input. We show that an estimator of the noise magnitude can be obtained from the root mean square (RMS) of the residual of OLS estimator. We further demonstrate the usefulness of removing such bias when estimating the Saupe tensor eigenvalue from homology fragments of ubiquitin, using RDCs measured in two different alignment medias. We note that there are other approaches to improve the estimation of Saupe tensor in the presence of structural noise by studying local bond orientations using multiple alignment medias [105, 142, 106, 122]. However in this work, we intend to remove the bias in the Saupe tensor eigenvalues in a single alignment media, when multiple Saupe tensor estimates is available from a collection of predetermined molecular fragments.

4.1 Notation

We summarize here the notation that is used in this particular chapter. Our notation is consistent with existing literature on RDC and Saupe tensor estimation. For a 3×3 matrix A, we use A_{ij}, $i, j = x, y, z$ to denote the nine entries of the matrix. When A is symmetric, we denote the eigendecomposition of A by

$$A = U(A)\Lambda(A)U(A)^T,$$

where $U(A)$ is an orthogonal matrix (i.e. $U(A)^T U(A) = U(A)U(A)^T = I_3$) and $\Lambda(A)$ is a diagonal matrix

$$\begin{bmatrix} \lambda_x(A) & 0 & 0 \\ 0 & \lambda_y(A) & 0 \\ 0 & 0 & \lambda_z(A) \end{bmatrix} \tag{4.8}$$

that contains the eigenvalues of A on the diagonal in ascending order. For a vector v, we often use $v(i)$ to denote its i-th entry, and $i = 1, \ldots, n$ if $v \in \mathbb{R}^n$. In the special case of $v \in \mathbb{R}^3$, we use v_x, v_y, v_z to denote each entry of the vector v. For a matrix A, we use A_i to denote its i-th column.

4.2 Debiasing Saupe tensor eigenvalues in the presence of structural noise

We now introduce a Monte Carlo method for correcting the bias in the eigenvalues of the OLS estimator arising from structural noise. For a protein with $N + 1$ peptide planes, we assume the $\{\phi_i, \psi_i\}_{i=1}^N$ torsion angles fully determine the backbone conformation. The template structure torsion angles ϕ_i^t, ψ_i^t's are related to the true structure via

$$\phi_i^t = \phi_i + \sigma\alpha_i, \qquad \psi_i^t = \psi_i + \sigma\beta_i, \quad i = 1, \ldots, N, \tag{4.9}$$

where α_i, β_i's are i.i.d. random normal variables with mean 0 and variance 1. Henceforth for a variable θ, we often make explicit the dependence on the torsion angles and noise by writing θ as $\theta(\phi_i^t, \psi_i^t)$. We also assume that the normalized dipolar coupling r is noiseless, i.e. $r = A(\phi_i, \psi_i)s$, where s corresponds to the entries of ground truth Saupe tensor S. The validity of this assumption is discussed in section 4.4. Our method consists of the following steps:

(1) Compute

$$\hat{s}(\phi_i^t, \psi_i^t) = (A(\phi_i^t, \psi_i^t)^T A(\phi_i^t, \psi_i^t))^{-1} A(\phi_i^t, \psi_i^t)^T r$$

.

(2) Generate n_1 copies of $A_{\text{sim}} = A(\phi_i^t + \sigma\alpha_i, \psi_i^t + \sigma\beta_i)$ by adding i.i.d. Gaussian noise with variance σ^2 to the torsion angles of the template structure.

(3) Find

$$\hat{s}_{\text{sim}} = \hat{s}(\phi_i^t + \sigma\alpha_i, \psi_i^t + \sigma\beta_i) = (A_{\text{sim}}^T A_{\text{sim}})^{-1} A_{\text{sim}}^T r.$$

(4) Let $\hat{S}$ and $\hat{S}_{\text{sim}}$ be the Saupe tensor estimators corresponding to $\hat{s}$ and $\hat{s}_{\text{sim}}$. Let

$$\widehat{\text{Bias}} = \langle \Lambda(\hat{S}_{\text{sim}}) \rangle_{\text{sim}} - \Lambda(\hat{S})$$

denote the bias estimate for the eigenvalues of the OLS estimator $\hat{S}$, where $\langle \cdot \rangle_{\text{sim}}$ denotes the averaging over n_1 simulated template structures. We propose using

$$\tilde{\Lambda} = \Lambda(\hat{S}) - \widehat{\text{Bias}} = 2\Lambda(\hat{S}) - \langle \Lambda(\hat{S}_{\text{sim}}) \rangle_{\text{sim}}$$

as an estimator with less bias.

The rationale of our method relies on the intuition that upon adding noise of similar magnitude to the linear system (1.10), the eigenvalues of the OLS estimator for the simulated samples should be biased away from $\Lambda(\hat{S})$ by an amount similar to difference

between $\Lambda(\hat{S})$ and the ground truth $\Lambda(S)$. This is also the intuition behind twicing [144], and related bootstrapped [49] biased reduced estimators. Alternatively, one can understand this procedure from the viewpoint of the SIMEX technique [37] for correcting bias resulting from regressor noise. Under SIMEX estimation framework one would simulate $A_{\text{sim}} = A(\phi_i^t + k\sigma\alpha_i, \psi_i^t + k\sigma\beta_i)$ with noise magnitudes of $k\sigma$ for various positive k to find out the dependency of $\Lambda(\hat{S}_{\text{sim}})$ on k. The $k = 0$ point corresponds to the case when no additional simulated noise is added, i.e. when the eigenvalue estimator is $\Lambda(\hat{S})$. From the extrapolation of the relation between $\Lambda(\hat{S}_{\text{sim}})$ and k one can obtain a debiased estimator at $k = -1$. Our method corresponds to the special case of SIMEX where we only add simulated noise with magnitude $k\sigma$ where $k = 1$. Our numerical results shows that this suffices for the application of Saupe tensor eigenvalue estimation.

4.2.1 Estimating noise level σ

We note that there is a caveat when using this parametric Monte Carlo method, in that it requires knowledge of the noise magnitude σ. Let the residual of the OLS estimator be defined as

$$e \equiv r - A\hat{s}.$$

In the simple case when additive noise with variance σ^2_{add} is added to the normalized dipolar couplings r, and A has no structural noise, i.e. $A = A(\phi_i, \psi_i)$, the dependence between the RMS of the residual, denoted RMS(e) and the noise magnitude can be readily calculated. In particular, an unbiased estimator of σ^2_{add} is given by [66]

$$\widehat{\sigma^2_{\text{add}}} = \frac{M}{M-5}\text{RMS}(e)^2.$$

Now in the case when there is noise on the design matrix $A = A(\phi_i^t, \psi_i^t)$ due to noise on the torsion angles (4.9), we show that there exists a linear dependence of RMS(e) on σ. We define $A_0 = A(\phi_i, \psi_i)$, and $A(\phi_i^t, \psi_i^t) = A_0 + E$. In this notation, normalized RDC

$r = A_0 s$. Then

$$\begin{aligned} \|e\|_2^2 &= \|r - A\hat{s}\|_2^2 \\ &= \|A_0 s - A(A^T A)^{-1} A^T (A_0 s)\|_2^2 \\ &= s^T A_0^T (I_M - A(A^T A)^{-1} A^T) A_0 s. \end{aligned} \tag{4.10}$$

The second equality follows from the fact that $I_M - A(A^T A)^{-1} A^T$ is a projection matrix. From

$$\begin{aligned} & A_0^T (I_M - A(A^T A)^{-1} A^T) A_0 \\ = \; & A_0^T A_0 - (A-E)^T A (A^T A)^{-1} A^T (A-E) \\ = \; & A_0^T A_0 - A^T A + E^T A + A^T E - E^T A (A^T A)^{-1} A^T E \\ = \; & A_0^T A_0 - (A-E)^T (A-E) + E^T E - E^T A (A^T A)^{-1} A^T E \\ = \; & E^T (I_M - A(A^T A)^{-1} A^T) E, \end{aligned} \tag{4.11}$$

we get

$$\begin{aligned} \|e\|_2^2 &= s^T E^T (I_M - A(A^T A)^{-1} A^T) E s \\ &\approx s^T E^T (I_M - A_0 (A_0^T A_0)^{-1} A_0^T) E s \\ &= s^T E^T P E s \end{aligned} \tag{4.12}$$

where $P = I_M - A_0 (A_0^T A_0)^{-1} A_0^T$ is a projection operator projecting vectors in $\mathbb{R}^M$ to $\mathbb{R}^{M-5}$. We drop the terms involving entries of E raised to the power greater than 2 to obtain the approximation in (4.12). Using Taylor expansion,

$$\begin{aligned} E_{ij} &= A_{ij} - A_{0ij} \\ &\approx \sum_{k=1}^{N} \left. \frac{\partial A_{ij}(\phi_k^t, \psi_k^t)}{\partial \phi_k^t} \right|_{\phi_k, \psi_k} \sigma \alpha_k + \left. \frac{\partial A_{ij}(\phi_k^t, \psi_k^t)}{\partial \psi_k^t} \right|_{\phi_k, \psi_k} \sigma \beta_k \\ &= F_{ij} \sigma. \end{aligned} \tag{4.13}$$

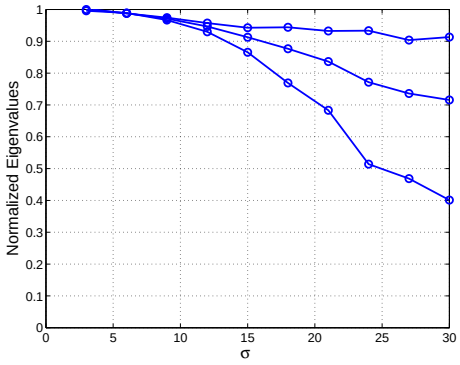


Figure 4.1: Plot of the eigenvalues of the OLS estimator $\hat{S}$ normalized by the eigenvalues of S v.s. σ. Increasing the noise level biases the eigenvalues towards zero. A fragment of Ubiquitin composed of 7 amino acids and a specific Saupe tensor S is used for the simulation and each point in the plot is computed from 200 different realizations of α_i, β_i's.

Plugging this into (4.12), it is clear that $\|e\|_2^2$ depends linearly on σ^2 and

$$\langle \mathrm{RMS}(e)^2 \rangle_{\alpha_i,\beta_i} \approx \frac{1}{M} \langle s^T F^T P F s \rangle_{\alpha_i,\beta_i} \sigma^2 \tag{4.14}$$

in the small noise regime. We therefore use

$$\hat{\sigma} = \sqrt{\frac{M}{s^T F^T P F s}} \mathrm{RMS}(e)$$

as the approximate noise magnitude when using the Monte Carlo method for bias reduction. Although we do not have the parameters s, F and P derived from the ground truth Saupe tensor and conformations, we can use $\hat{s}$ as surrogate of s, and use the noisy structure to derive an approximation of F and P.

4.3 Numerical results

We first demonstrate that $\hat{\sigma}$ obtained through the simulation method in section 4.2.1 is a good estimate of σ. For simulation purposes, we use a segment of Ubiquitin with 7 amino acids containing 21 $N-H$, $C-CA$ and $C-N$ bonds. In Fig. 4.2(left), we plot

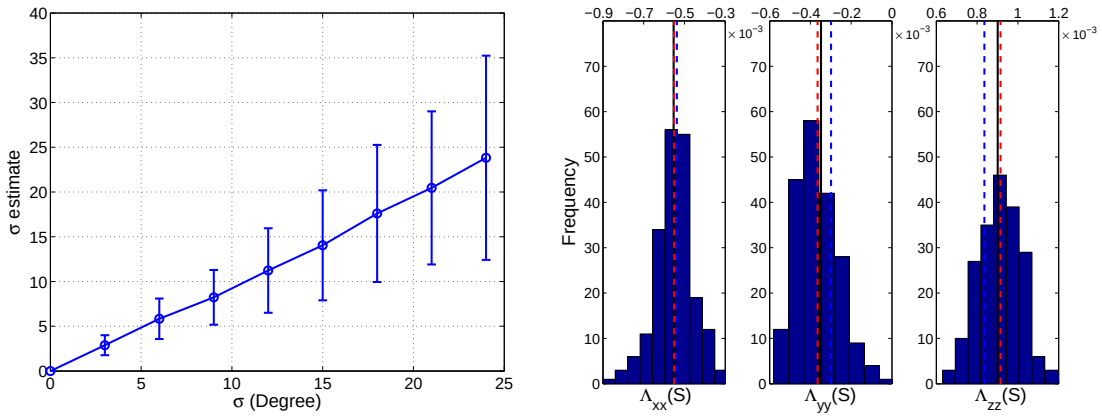


Figure 4.2: **Left:** Plot of $\hat{\sigma}$ v.s. σ. For a given noise level σ, $\hat{\sigma}$ is averaged over 200 different realizations of α_i, β_i's. **Right:** Histograms of the diagonal entries of $\Lambda(\hat{S})$ and $\tilde{\Lambda}$ obtained from 200 fragment conformations with 20° noise on the torsion angles. The values of $\Lambda(S)$, $\langle \Lambda(\hat{S}) \rangle_{\alpha_i,\beta_i}$ and $\langle \tilde{\Lambda} \rangle_{\alpha_i,\beta_i}$ are denoted by black, blue and red line respectively.

$\hat{\sigma}$ v.s. σ. For each value of σ we calculate $\hat{\sigma}$ for 200 realizations of S and $\{\alpha_i, \beta_i\}_{i=1}^6$. For a protein fragment with 7 residues, we let $n_2 = 200$ in step (2) of the σ estimation procedure in section 4.2.1. The simulation shows that $\hat{\sigma}$ is rather close to σ, especially when the angular noise is less than 12 degrees.

We next show that the SIMEX-like method proposed in section 4.2 is able to reduce the bias in eigenvalue estimation, where the bias of an estimator $\hat{\theta}$ of parameter θ is defined to be

$$\text{Bias}(\hat{\theta}) = \langle \hat{\theta} \rangle - \theta.$$

$\langle \cdot \rangle$ denotes averaging over the distribution of data. For this simulation, we use a specific ground truth Saupe tensor and the aforementioned Ubiquitin fragment to generate clean RDC measurements. From the fragment, 200 realizations of noisy conformation are generated with $\sigma = 20°$. To obtain $\tilde{\Lambda}$, we set $n_1 = 8000$ when simulating A_b in step (2) of the Monte Carlo procedure. In Fig. 4.2(right), we see that the values of $\langle \tilde{\Lambda} \rangle_{\alpha_i,\beta_i}$ (Red dotted line) obtained from averaging over 200 samples are almost the same as the eigenvalues of S (Black line), while there is a clear bias in the estimator $\Lambda(\hat{S})$ (Blue dotted line).

4.3.1 Application: Saupe tensor estimation from multiple molecular fragments

While the proposed eigenvalue estimator $\tilde{\Lambda}$ has less bias, it is not necessary that $\tilde{\Lambda}$ has a lower mean squared error (MSE). This can be understood from the *bias-variance decomposition*, which is a classical way in statistics to decompose the MSE of an estimator $\hat{\theta}$. The MSE of an estimator $\hat{\theta}$ admits the following decomposition

$$\begin{aligned} \mathrm{MSE}(\hat{\theta}) &= \langle(\theta - \hat{\theta})^2\rangle \\ &= \langle(\theta - \langle\hat{\theta}\rangle + \langle\hat{\theta}\rangle - \hat{\theta})^2\rangle \\ &= \mathrm{Bias}(\hat{\theta})^2 + \mathrm{Var}(\hat{\theta}) + 2(\theta - \langle\hat{\theta}\rangle)\langle\langle\hat{\theta}\rangle - \hat{\theta}\rangle \\ &= \mathrm{Bias}(\hat{\theta})^2 + \mathrm{Var}(\hat{\theta}). \end{aligned} \tag{4.15}$$

$\mathrm{Var}(\hat{\theta})$ denotes the variance of $\hat{\theta}$. Although we achieve less bias with the estimator $\tilde{\Lambda}$, we pay the price of having larger variance due to bias estimation involved in obtaining $\tilde{\Lambda}$. This increase in variance can lead to $\tilde{\Lambda}$ having higher MSE than $\Lambda(\hat{S})$. From this point of view, when estimating the Saupe tensor eigenvalue using a single template fragment, the Monte Carlo method for debiasing may seem unnecessary or even disadvantageous. However, when multiple template fragments are available, the average of $\tilde{\Lambda}$ over these fragments, denoted $\tilde{\Lambda}_{\mathrm{ave}}$, enjoys variance reduction proportional to the number of fragments. Therefore in the case when there are many fragments, it is worth paying the price of increased variance because the systematic bias error cannot be reduced via averaging. In the rest of the section, we use $\Lambda_{\mathrm{ave}}(\hat{S})$ to denote the average of $\Lambda(\hat{S})$ over multiple fragments.

We now demonstrate the usefulness of our method under the setting of Molecular Fragment Replacement (MFR) [43, 92]. When RDCs are measured in two different alignment medias for a protein of unknown structure, the MFR method can construct its structure by combining short homologous fragments obtained from chemical shift and

dipolar homology database mining. Typically for every protein fragment of 7 residues, 10 homologous structures are searched based on the similarity of chemical shifts and the goodness of Saupe tensor fit to the observed RDC. When OLS is used to fit the Saupe tensor with design matrix A constructed from homologous structures, one can average all OLS eigenvalue estimated to obtain improved estimators of the parameters such as alignment magnitude and rhombicity that depend on the eigenvalues [92]. These parameters can in turn be used in a simulated annealing procedure such as XPLOR-NIH [128] to refine the structure.

We first use synthetic data to demonstrate our method. We generate 12 random Saupe tensors, by sampling two eigenvalues from the uniform distribution on $[-10^{-3}, 0]$ and $[0, 10^{-3}]$ respectively, and extract the third eigenvalue by requiring $\Lambda(S)_{xx}+\Lambda(S)_{yy}+\Lambda(S)_{zz}=0$. The orthogonal matrix $U(S)$ is sampled uniformly from the group of 3×3 orthogonal matrices, by computing the orthogonal factor in the polar decomposition of a 3×3 Gaussian random matrix [18]. After obtaining the RDC d_{nm}'s from the clean structure and the ground truth Saupe tensor, under each simulated alignment condition we add structural noise of magnitude σ to every fragment of 7 amino acids of the Ubiquitin structure obtained from X-ray crystallography (PDB ID 1UBQ). We evaluate the estimators of the Saupe tensor eigenvalues $\Lambda_{\text{ave}}(\hat{S})$ and $\tilde{\Lambda}_{\text{ave}}$ computed from the average of $\Lambda(\hat{S})$ and $\tilde{\Lambda}$ of all fragments, by comparing their fractional errors averaged over the 12 different Saupe tensors and torsion angle noise realizations in Fig. 4.3. The fractional error is defined as

$$\frac{\|\Lambda_{\text{ave}}(\hat{S})-\Lambda(S)\|_F}{\|\Lambda(S)\|_F} \quad \text{and} \quad \frac{\|\tilde{\Lambda}_{\text{ave}}-\Lambda(S)\|_F}{\|\Lambda(S)\|_F}.$$

In this simulation, the fractional error of $\Lambda_{\text{ave}}(\hat{S})$ is at least three times larger than $\tilde{\Lambda}_{\text{ave}}$.

We finally apply this method to estimate the Saupe tensor of ubiquitin in two different alignment medias using the experimental RDC data in [38]. From 600 homologous

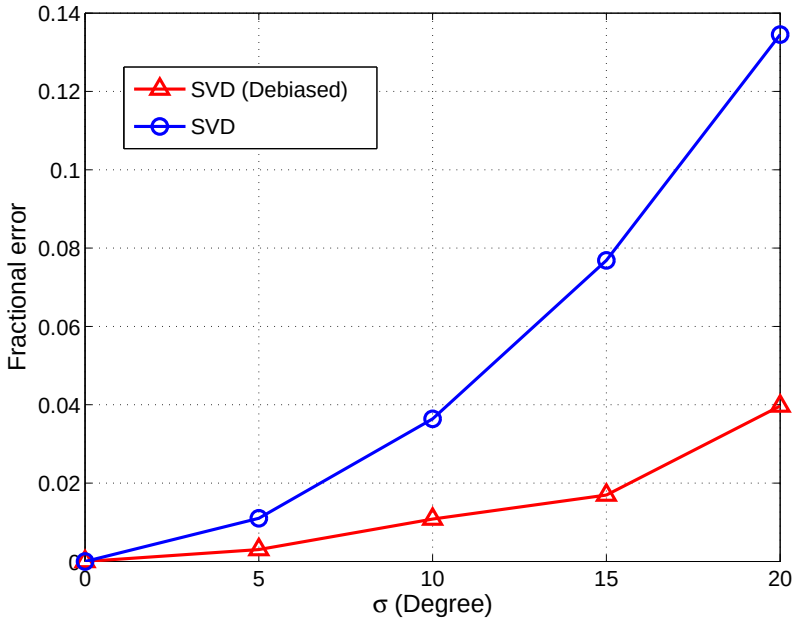


Figure 4.3: Plot of the fractional error of $\Lambda_{\text{ave}}(\hat{S})$ and $\tilde{\Lambda}_{\text{ave}}$ v.s. σ. Each data point is averaged over 12 different Saupe tensor and noise realizations for 1UBQ. The plot shows a clear advantage of the bias reduced estimator over the OLS estimator.

structures returned by MFR homology search, each containing 7 residues, we obtain 600 Saupe tensor estimates using OLS. Since we expect our method to have a significant effect for fragments severely corrupted by structural noise, we average the fragments with residual RMS *above* a certain threshold and plot $\Lambda_{\text{ave}}(\hat{S})$ and $\tilde{\Lambda}_{\text{ave}}$ normalized by $\Lambda(S)$ v.s. RMS thresholds. To get an approximate ground truth Saupe tensor S, we use the high resolution Ubiquitin structure 1UBQ obtained from X-ray crystallography [147] to fit the RDC data. We demonstrate the results in Fig. 4.4. Other than the estimators for $\Lambda(S)_{yy}$ of the second alignment media which has a large percent error due to the relatively small magnitude of $\Lambda(S)_{yy}$, $\tilde{\Lambda}_{\text{ave}}$ typically achieves 0.9 of the ground truth value, whereas $\Lambda_{\text{ave}}(\hat{S})$ can shrink to 0.8 of the value of $\Lambda(S)$ when only the fragments of high RMS are used in averaging. We therefore recommend the use of our proposed bias removing method when estimating eigenvalues from multiple noisy fragments.

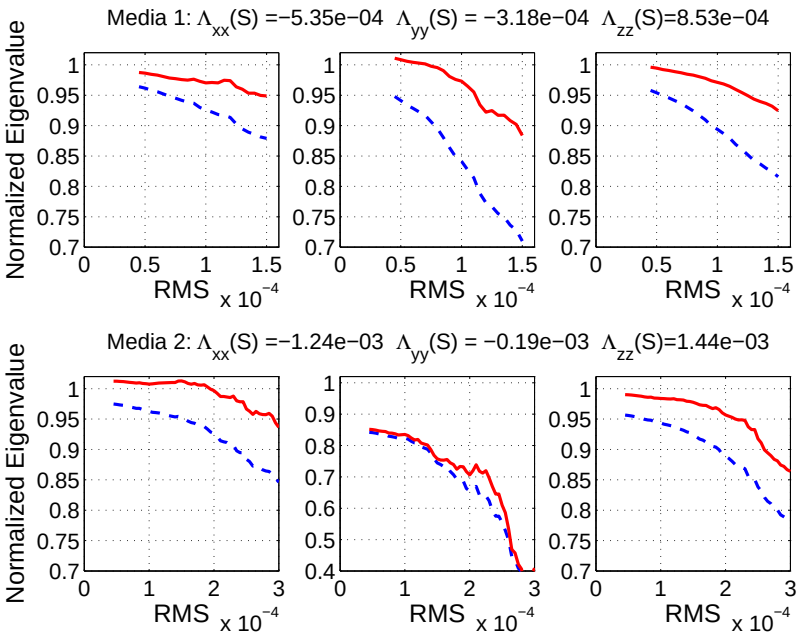


Figure 4.4: Plot of the eigenvalue estimators normalized by $\Lambda(S)$ v.s. residual RMS thresholds. Estimators are obtained from experimental RDC measurements in two different alignment medias. While the magnitude of $\tilde{\Lambda}_{\text{ave}}$ (Red curves) and $\Lambda_{\text{ave}}(\hat{S})$ (Blue dotted curves) both decrease as low quality (high RMS) fragments are solely used in averaging, $\tilde{\Lambda}_{\text{ave}}$ in general is within 90% of the ground truth value but $\Lambda(S)$ drops to 80% of $\Lambda_{\text{ave}}(\hat{S})$. The value of $\Lambda(S)$ for both alignment medias are indicated in the plot title.

4.4 Additive measurement noise on RDC v.s. structural noise

So far we have been neglecting the presence of additive noise on r_{nm}, which is considered by [101]. We define the noisy RDC measurements corrupted by additive noise as

$$r_{\text{add}} = r + \sigma_{\text{add}}\varepsilon \tag{4.16}$$

where entries of the column vector ε are i.i.d random variables with mean zero. In this section, we show using perturbation theory that this type of additive noise biases the eigenvalue magnitude positively, therefore it cannot explain the magnitude shrinkage we see when fitting the Saupe tensor to real RDC data (Fig. 4.4). Moreover, the order of magnitude of this positive bias is not sufficient to explain the error between $\hat{S}$ and S. This has been noted by the authors of [101] that in order to account for the size of the OLS misfit, an uncertainly of 2-3 Hz for the RDC measurements is required although the experimental uncertainty is only about 0.2-0.5 Hz. This is the reason why in this chapter we focus on removing the bias that arises from structural noise.

Let

$$S = U(S)\Lambda(S)U(S)^T \tag{4.17}$$

be the eigendecomposition of S. Assuming the eigenvalues of S are nondegenerate, the second order perturbation theory [95] states

$$\begin{aligned}\lambda_j(\hat{S}) \;\approx\; & \lambda_j(S) + U(S)_j^T(\hat{S}-S)U(S)_j \\ & + \sum_{\substack{k=x,y,z,\\ k\neq j}} \frac{(U(S)_k^T(\hat{S}-S)U(S)_j)^2}{\lambda_j(S)-\lambda_k(S)}.\end{aligned} \tag{4.18}$$

Averaging the perturbation expansion over the distribution of ε, we get

$$\lambda_j(\hat{S}) \approx \lambda_j(S) + \sum_{\substack{k=x,y,z,\\ k\neq j}} \frac{\langle (U(S)_k^T(\hat{S}-S)U(S)_j)^2\rangle_\varepsilon}{\lambda_j(S)-\lambda_k(S)}. \tag{4.19}$$

Here we use the fact that $\langle \hat{S}-S\rangle_\varepsilon = 0$ since

$$\langle \hat{s}-s\rangle_\varepsilon = \langle (A^TA)^{-1}A^T(As+\varepsilon)-s\rangle_\varepsilon = \langle (A^TA)^{-1}A^T\varepsilon\rangle_\varepsilon = 0$$

The expression in (4.19) reveals that in the presence of noise, the largest eigenvalue of $\hat{S}$ is always greater than the largest eigenvalue of S, while the smallest eigenvalue behaves in the exact opposite manner. Such effect of bias of pushing the extreme eigenvalues outwards is also commonly seen in the context of estimating the extreme eigenvalues of covariance matrices [126].

For this type of bias we now give an estimate of its order of magnitude. First we bound the numerator in the second order correction term in (4.19):

$$\begin{aligned}(U(S)_k^T(\hat{S}-S)U(S)_j)^2 &= \mathrm{Tr}((\hat{S}-S)U(S)_jU(S)_k^T)^2\\ &\leq \|\hat{S}-S\|_F^2\|U(S)_jU(S)_k^T\|_F^2\\ &\leq 3\|\hat{s}-s\|_2^2. \end{aligned}\tag{4.20}$$

The first inequality results from Cauchy-Schwarz inequality, and the second inequality relies on the fact that $\|U(S)_jU(S)_k^T\|_F = 1$ and $\|\hat{S}-S\|_F^2 \leq 3\|\hat{s}-s\|_2^2$, which can be verified easily. It is a classical result [66] that the OLS estimator has covariance matrix

$$\langle (\hat{s}-s)(\hat{s}-s)^T\rangle_\varepsilon = \sigma_{\text{add}}^2(A^TA)^{-1}, \tag{4.21}$$

therefore

$$\langle \|\hat{s}-s\|_2^2\rangle_\varepsilon = \sigma_{\text{add}}^2\mathrm{Tr}((A^TA)^{-1}). \tag{4.22}$$

Now we make some assumptions in order to derive a "guesstimate" of $\mathrm{Tr}((A^TA)^{-1})$. We assume that the eigenvalues of A^TA are similar to each other, which implies

$$\lambda(A^TA) \approx \mathrm{Tr}(A^TA)/5 = \mathrm{Tr}(AA^T)/5. \tag{4.23}$$

In such case from Eq. (4.22) we get

$$\langle \|\hat{s}-s\|_2^2 \rangle_\varepsilon \approx \sigma_{\mathrm{add}}^2 \frac{5}{\lambda(A^TA)} = \frac{25\sigma_{\mathrm{add}}^2}{\mathrm{Tr}(AA^T)}. \tag{4.24}$$

Here we locally denote A_i as the i-th row of A. If we assume M is sufficiently large, by law of large number

$$\begin{aligned} \frac{1}{M}\mathrm{Tr}(AA^T) &= (1/M)\sum_{i=1}^{M} \|A_i\|_2^2 \\ &\approx \langle (v_x^2-v_y^2)^2 + (v_x^2-v_z^2)^2 + (2v_xv_y)^2 + (2v_yv_z)^2 + (2v_xv_z)^2 \rangle_v, \end{aligned} \tag{4.25}$$

where $\langle\cdot\rangle_v$ denotes the averaging over the distribution of bond direction v. Assuming the bond directions distribute uniformly over the unit sphere in $\mathbb{R}^3$,

$$\begin{aligned} & \langle (v_x^2-v_y^2)^2 + (v_x^2-v_z^2)^2 + (2v_xv_y)^2 + (2v_yv_z)^2 + (2v_xv_z)^2 \rangle_v \\ = \;& 2\langle (v_x^2-v_y^2)^2 \rangle_v + 3\langle (2v_xv_y)^2 \rangle_v \\ = \;& \int_0^{2\pi}\int_0^{\pi} 2\sin^4(\gamma)(\sin^2(\kappa)-\cos^2(\kappa))^2 \\ & \qquad +3\sin^4(\gamma)(2\cos(\kappa)\sin(\kappa))^2 \; \frac{\sin(\gamma)d\gamma d\kappa}{4\pi} \\ = \;& \frac{4}{3}. \end{aligned} \tag{4.26}$$

Based on these assumptions on the bond directions, we finally derive an estimate

$$\langle \|\hat{s}-s\|_2^2 \rangle_\varepsilon \approx \frac{75}{4M}\sigma_{\mathrm{add}}^2 \tag{4.27}$$

by combining (4.24) and (4.26). We put this estimate into (4.20) and obtain an upper-bound for the bias in (4.19). Taking $\lambda_z(\hat{S})$ for example:

$$\lambda_z(\hat{S}) - \lambda_z(S) \lesssim \frac{1}{\lambda_z(S) - \lambda_y(S)} \frac{225}{4M} \sigma_{\text{add}}^2 \tag{4.28}$$

We now give an estimate of the order of magnitude of the bias. Since the magnitude of the extreme eigenvalues of the Saupe tensor is around 10^{-3}, for example for the two RDC datasets acquired for Ubiquitin, we simply assume $\lambda_z(S) - \lambda_y(S) \sim 10^{-4}$. The typical experimental uncertainty for RDC measurements is about 0.2 Hz - 0.5 Hz, and the dipolar coupling constant D_0 for e.g. $N-H$ bonds, is about 23 kHz, therefore the noise magnitude σ_{add} of the additive noise on the normalized dipolar coupling is about $0.5/(23 \times 10^3) \approx 2 \times 10^{-5}$. For a fragment of 7 amino acid, we have $M = 21$ RDC measurements. Plugging these numbers into (4.28), we get

$$\lambda_z(\hat{S}) - \lambda_z(S) \lesssim 10^{-5},$$

which amounts to 1% error when $\lambda_z(S) \sim 10^{-3}$. This cannot explain the 10% or larger error in fitting Saupe tensor to real RDC datasets using homology fragments in the previous section.

We present a simulation to illustrate the bias in OLS eigenvalues estimation in the presence of additive noise. We use the Saupe tensor eigenvalues for Ubiquitin in the first alignment media presented in Fig. 4.4, and a Ubiquitin fragment consisting of 7 amino acids for this simulation. We generate noisy datasets using the noise model

$$r_{\text{add}} = As + \sigma_{\text{add}} \varepsilon.$$

For every noise level, we average $\Lambda(\hat{S})$ normalized by $\Lambda(S)$ over 500 different realizations of s and ε where entries of ε are i.i.d. random normal variables. The different realization

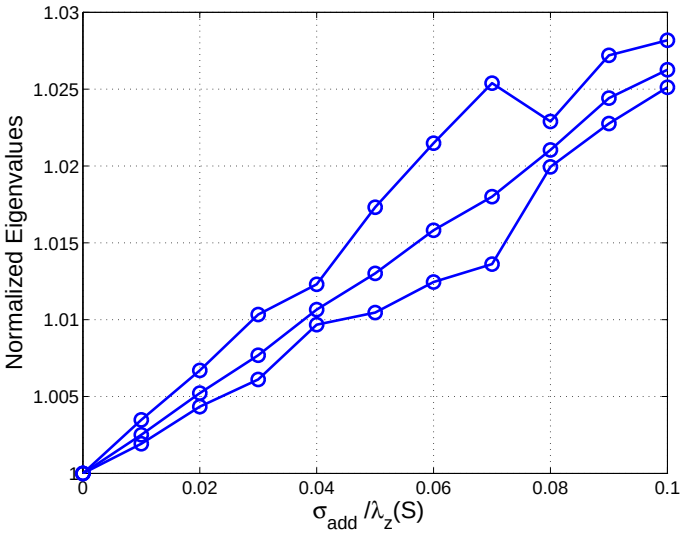


Figure 4.5: Plot of the three eigenvalues of the OLS estimator $\hat{S}$ normalized by the eigenvalues of S v.s. σ_{add} under noise model (4.16). Each point is averaged over 500 noise and Saupe tensor realizations. Increasing the noise level biases the eigenvalues positively, unlike the case for structural noise. At 10% noise level, the bias is about 3%.

of s are generated from $S = U(S)\Lambda(S)U(S)^T$ where $\Lambda(S)$ is fixed but $U(S)$ is sampled uniformly from the orthogonal group in $\mathbb{R}^3$. We vary the orientation of the Saupe tensor since it is clear from (4.19) that the bias depends on $U(S)$. We change σ_{add} from 0 to 10% of $\lambda_z(S)$ and present the results in Fig. 4.5. We note again from previous calculations, $\sigma_{\text{add}} \sim 2 \times 10^{-5}$, which amounts to 2-3% of the $\lambda_z(S) = 0.85 \times 10^{-3}$ considered. As shown in the simulation and our crude estimate, such magnitude of noise gives rise to bias error of about 1%. Even in the case of having very noisy RDC (having noise magnitude 10% of $\lambda_z(S)$), the bias error caused by additive noise is around 3%. Whereas in a typical MFR search with torsion angle tolerance being set to $\pm 20^\circ - 30^\circ$ [164], the simulation in Fig. 4.1 suggests structural noise can cause bias error sometimes much greater than 10%. Therefore in this chapter we focus on removing the bias that arises from structural noise. In the case when accurate template structure is available and the additive noise is a concern, we refer readers to the appendix for the removal of such bias using an analytic expression derived from perturbation theory.

4.5 Conclusion

We observe a negative bias when estimating the Saupe tensor eigenvalues through the classical SVD method, in the presence of structural noise on the template structure due to torsion angle noise. We present a Monte Carlo method that simulates noise on the template structure by perturbing the torsion angles and use the simulated structure to estimate the bias in the eigenvalues. We demonstrate the effectiveness of our method in reducing the error arising from bias when estimating Saupe tensor eigenvalues from multiple protein fragments, which is a natural setting to consider when building protein structure from homologous substructures.

4.6 Appendix

4.6.1 Removing bias from additive noise

Define a linear operator $L : \mathbb{R}^5 \to \mathbb{R}^{3\times 3}$ that forms a Saupe tensor S from the vector s as

$$L(s) = \begin{bmatrix} -s(1)-s(2) & s(3) & s(4) \\ s(3) & s(1) & s(5) \\ s(4) & s(5) & s(2) \end{bmatrix}, \quad s \in \mathbb{R}^5. \tag{4.29}$$

For the additive noise model (4.16) we have

$$\hat{S} = L(\hat{s}) = L((A^TA)^{-1}A^T r_{\text{add}}) = S + L((A^TA)^{-1}A^T\varepsilon). \tag{4.30}$$

We also define the adjoint operator of L, $L^* : \mathbb{R}^{3\times 3} \to \mathbb{R}^5$ through the relation

$$\text{Tr}(X^T L(y)) = L^*(X)^T y, \tag{4.31}$$

for every $y \in \mathbb{R}^5$ and $X \in \mathbb{R}^{3\times 3}$. To obtain the form of L^*, we let $y \in \{e_1, \ldots, e_5\}$, where $e_i(i) = 1$ and $e_i(j) = 0$ if $j \neq i$. Plugging such y into Eq. (4.31), we get

$$\begin{aligned} L^*(X)(1) &= -X_{xx} + X_{yy} \\ L^*(X)(2) &= -X_{xx} + X_{zz} \\ L^*(X)(3) &= X_{xy} + X_{yx} \\ L^*(X)(4) &= X_{xz} + X_{zx} \\ L^*(X)(5) &= X_{yz} + X_{zy} \end{aligned} \tag{4.32}$$

Using such notion of the adjoint operator, the perturbation series in (4.19) can be written as

$$\begin{aligned} \langle \lambda_j(\hat{S}) \rangle_\varepsilon &\approx \lambda_j(S) + \sum_{\substack{k=x,y,z, \\ k\neq j}} \frac{\langle ((\hat{s}-s)^T L^*(U(S)_k U(S)_j^T))^2 \rangle_\varepsilon}{\lambda_j(S) - \lambda_k(S)} \\ &= \lambda_j(S) + \mathrm{Tr}\Bigg[\Bigg(\sum_{\substack{k=x,y,z, \\ k\neq j}} \frac{L^*(U(S)_k U(S)_j^T) L^*(U(S)_k U(S)_j^T)^T}{\lambda_j(S) - \lambda_k(S)} \Bigg) \mathrm{Var}(\hat{s}) \Bigg] \end{aligned} \tag{4.33}$$

where

$$\mathrm{Var}(\hat{s}) \equiv \langle (\hat{s}-s)(\hat{s}-s)^T \rangle_\varepsilon = (A^T A)^{-1} \sigma^2_{\text{add}}. \tag{4.34}$$

Therefore we can subtract the second order term in (4.33) to correct for the bias in the eigenvalues. Although we do not know the eigenvectors and eigenvalues of S, we can replace them with the eigenvectors and eigenvalues $\hat{S}$. This change will only affect on the higher order terms in the perturbation series.

Chapter 5

Conclusion and outlook

In this thesis, we investigate the structural calculation problem arises from NMR spectroscopy. We bring improvement in terms of speed and accuracy in solving the non-convex problems encountered using convex optimization. In particular, we propose a scalable divide-and-conquer approach to solve the distance geometry problem via global registration. Furthermore, we provide the first convex programming approach to calculate protein structure from NOE and RDC in an integrated manner. However, only towards the end of my doctoral study I realized there is much yet to be studied. Therefore instead of giving a conclusion, here we focus on giving a summary on a few problems we want to tackle. The summary of the technical contributions can be found in each chapter.

The main difficulty of using NMR to determine the protein structure with weight larger than 25 kDa is the degradation of signal to noise ratio [53]. Slower tumbling of large molecules lead to enhanced spin-spin interaction, leading to faster relaxation of magnetization. The direct result of fast relaxation is the broadening of the line shape of resonance peaks. This issue is exacerbated by the large number of nuclei in large protein which gives rise to many overlapping resonances. The expert knowledge of a NMR practitioner is generally needed to manually filter the resonance peaks and perform spectral assignment iteratively before structural calculation. Current automated

spectral assignment procedures [3, 83, 127] are multi-step in nature, therefore allowing error to accumulate and information to be lost easily. Hence from a computational perspective, we hope to provide NMR spectroscopist with an automated procedure for spectral assignment that uses NMR spectra in a direct manner, notwithstanding wrong identification of peaks or missing peaks.

The challenge of spectral assignment can also be addressed with a more versatile structural calculation algorithm that can work with various sets of data. In general it is more difficult to obtain unambiguous spectral assignment of the side-chain atoms. This leads to ambiguity in NOE assignment, especially for larger system. RDC on the other hand provides a way to bypass NOE assignment to obtain a protein backbone conformation. The large magnitude of N-H and CA-HA RDC can be detected rather easily even for large protein, especially in deuterated protein sample [154]. The spectral assignment can be validated easily if an accurate ab-initio structure of the backbone can be derived from minimal set of RDC and backbone NOE. We also want to extend our methods to deal with database derived restraints. For example, torsion angle restraints can be derived from chemical shifts of backbone atoms using TALOS [130]. Furthermore, side-chain rotamer library [104] can be used to model protein side-chains.

In the end, we remark that NMR spectroscopy is not defined by a single set of experiment. Many different interactions can be measured with varying experimental cost. Our final goal will be to design a complete pipeline for spectral assignment and structural calculation that uses an optimally chosen set of measurements.

Bibliography

[1] E. Abbe, A. S. Bandeira, and G. Hall, *Exact recovery in the stochastic block model*, arXiv preprint arXiv:1405.3267 (2014).

[2] P.-A. Absil, R. Mahony, and R. Sepulchre, *Optimization Algorithms on Matrix Manifolds*, Princeton University Press, 2009.

[3] B. Alipanahi, X. Gao, E. Karakoc, S. C. Li, F. Balbach, G. Feng, L. Donaldson, and M. Li, *Error tolerant NMR backbone resonance assignment and automated structure generation*, Journal of bioinformatics and computational biology **9** (2011), no. 01, 15–41.

[4] B. Alipanahi, N. Krislock, A. Ghodsi, H. Wolkowicz, L. Donaldson, and M. Li, *Determining protein structures from NOESY distance constraints by semidefinite programming*, Journal of Computational Biology **20** (2013), no. 4, 296–310.

[5] S. L. Altmann, *Rotations, quaternions, and double groups*, Courier Corporation, 2005.

[6] M. Andriluka, S. Roth, and B. Schiele, *Pictorial structures revisited: People detection and articulated pose estimation*, Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, IEEE, 2009, pp. 1014–1021.

[7] K. Anstreicher and H. Wolkowicz, *On Lagrangian relaxation of quadratic matrix constraints*, SIAM Journal on Matrix Analysis and Applications **22** (2000), no. 1, 41–55.

[8] K. S. Arun, T. S. Huang, and S. D. Blostein, *Least-squares fitting of two 3d point sets*, IEEE Transactions on Pattern Analysis and Machine Intelligence (1987), no. 5, 698–700.

[9] A. S. Bandeira, , A. Singer, and D. A. Spielman, *A cheeger inequality for the graph connection laplacian*, arXiv:1204.3873 (2012).

[10] A. S. Bandeira, C. Kennedy, and A. Singer, *Approximating the little Grothendieck problem over the orthogonal group*, arXiv:1308.5207 (2013).

[11] A. Bax, G. Kontaxis, and N. Tjandra, *Dipolar couplings in macromolecular structure determination.*, Methods in enzymology **339** (2001), 127.

[12] S. R. Becker, E. J. Candès, and M. C. Grant, *Templates for convex cone problems with applications to sparse signal recovery*, Mathematical Programming Computation **3** (2011), no. 3, 165–218.

[13] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov, and P. E. Bourne, *The protein data bank*, Nucleic acids research **28** (2000), no. 1, 235–242.

[14] D. P. Bertsekas, *Nonlinear programming*, (1999).

[15] P. J. Besl and N. D. McKay, *A method for registration of 3d shapes*, IEEE Transactions on Pattern Analysis and Machine Intelligence **14** (1992), no. 2, 239–256.

[16] R. Bhatia, *Matrix Analysis*, vol. 169, Springer, 1997.

[17] P. Biswas, T-C Liang, K-C Toh, Y. Ye, and T-C Wang, *Semidefinite programming approaches for sensor network localization with noisy distance measurements*, Automation Science and Engineering, IEEE Transactions on **3** (2006), no. 4, 360–371.

[18] G. Blower, *Random matrices: high dimensional phenomena*, vol. 367, Cambridge University Press, 2009.

[19] A. M. Bonvin and A. T. Brünger, *Do NOE distances contain enough information to assess the relative populations of multi-conformer structures?*, Journal of biomolecular NMR **7** (1996), no. 1, 72–76.

[20] I. Borg and P. J. Groenen, *Modern multidimensional scaling: Theory and applications*, Springer Science & Business Media, 2005.

[21] N. Boumal, B. Mishra, P-A Absil, and R. Sepulchre, *Manopt, a matlab toolbox for optimization on manifolds*, The Journal of Machine Learning Research **15** (2014), no. 1, 1455–1459.

[22] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, *Distributed optimization and statistical learning via the alternating direction method of multipliers*, Foundations and Trends® in Machine Learning **3** (2011), no. 1, 1–122.

[23] S. Boyd and L. Vandenberghe, *Convex optimization*, Cambridge university press, 2004.

[24] M. Bryson, F. Tian, J. H. Prestegard, and H. Valafar, *REDCRAFT: a tool for simultaneous characterization of protein backbone structure and motion from RDC data*, Journal of Magnetic Resonance **191** (2008), no. 2, 322–334.

[25] S. Burer and R. D. C. Monteiro, *A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization*, Mathematical Programming **95** (2003), no. 2, 329–357.

[26] E. Candes and B. Recht, *Exact matrix completion via convex optimization*, Foundations of Computational Mathematics **9** (2009), no. 6, 717–772.

[27] E. J. Candes, T. Strohmer, and V. Voroninski, *Phaselift: Exact and stable signal recovery from magnitude measurements via convex programming*, Communications on Pure and Applied Mathematics **66** (2013), no. 8, 1241–1274.

[28] R. J. Carroll, D. Ruppert, L. A. Stefanski, and C. M. Crainiceanu, *Measurement error in nonlinear models: a modern perspective*, CRC press, 2006.

[29] G. Casella and R. L. Berger, *Statistical inference*, vol. 2, Duxbury Pacific Grove, CA, 2002.

[30] A. Cassioli, B. Bardiaux, G. Bouvier, A. Mucherino, R. Alves, L. Liberti, M. Nilges, C. Lavor, and T. Malliavin, *An algorithm to enumerate all possible protein conformations verifying a set of distance constraints*, BMC bioinformatics **16** (2015), no. 1, 23.

[31] I. R. Chandrashekar, A. Dike, and S. M. Cowsik, *Membrane-induced structure of the mammalian tachykinin Neuropeptide gamma*, Journal of structural biology **148** (2004), no. 3, 315–325.

[32] K. N. Chaudhury, Y. Khoo, and A. Singer, *Global registration of multiple point clouds using semidefinite programming*, SIAM Journal on Optimization **25** (2015), no. 1, 468–501.

[33] K. Chen and N. Tjandra, *The use of residual dipolar coupling in studying proteins by NMR*, NMR of Proteins and Small Biomolecules, Springer, 2012, pp. 47–67.

[34] F. R. K. Chung, *Spectral Graph Theory*, vol. 92, American Mathematical Society, 1997.

[35] G. M. Clore, A. M. Gronenborn, and N. Tjandra, *Direct structure refinement against residual dipolar couplings in the presence of rhombicity of unknown magnitude*, Journal of Magnetic Resonance **131** (1998), no. 1, 159–162.

[36] Robert Connelly, *Rigidity and energy*, Inventiones Mathematicae **66** (1982), no. 1, 11–33.

[37] J. R. Cook and L. A. Stefanski, *Simulation-extrapolation estimation in parametric measurement error models*, Journal of the American Statistical Association **89** (1994), no. 428, 1314–1328.

[38] G. Cornilescu, J. L. Marquardt, M. Ottiger, and A. Bax, *Validation of protein structure from anisotropic carbonyl chemical shifts in a dilute liquid crystalline phase*, Journal of the American Chemical Society **120** (1998), no. 27, 6836–6837.

[39] G. M. Crippen and T. F. Havel, *Distance geometry and molecular conformation*, vol. 74, Research Studies Press Taunton, England, 1988.

[40] M. Cucuringu, Y. Lipman, and A. Singer, *Sensor network localization by eigenvector synchronization over the euclidean group*, ACM Transactions on Sensor Networks **8** (2012), no. 3, 19.

[41] M. Cucuringu, A. Singer, and D. Cowburn, *Eigenvector synchronization, graph rigidity and the molecule problem*, Information and Inference **1** (2012), no. 1, 21–67.

[42] M. Cucuringu, A. Singer, and D. Cowburn, *Eigenvector synchronization, graph rigidity and the molecule problem*, Information and Inference **1** (2012), no. 1, 21–67.

[43] F. Delaglio, G. Kontaxis, and A. Bax, *Protein structure determination using molecular fragment replacement and NMR dipolar couplings*, Journal of the American Chemical Society **122** (2000), no. 9, 2142–2143.

[44] M. Dodig, M. Stošić, and J. Xavier, *On minimizing a quadratic function on Stiefel manifold*, Linear Algebra and its Applications **475** (2015), 251–264.

[45] B. R. Donald, *Algorithms in structural molecular biology*, MIT Press Cambridge, MA:, 2011.

[46] M. Dür, *Copositive programming–a survey*, Recent advances in optimization and its applications in engineering, Springer, 2010, pp. 3–20.

[47] A. Edelman, T. A. Arias, and S. T. Smith, *The geometry of algorithms with orthogonality constraints*, SIAM Journal on Matrix Analysis and Applications **20** (1998), no. 2, 303–353.

[48] M. Edén, *Zeeman truncation in NMR. i. the role of operator commutation*, Concepts in Magnetic Resonance Part A **43** (2014), no. 4, 91–108.

[49] B. Efron and R. J. Tibshirani, *An introduction to the bootstrap*, CRC press, 1994.

[50] K. Fan and A. J. Hoffman, *Some metric inequalities in the space of matrices*, Proceedings of the American Mathematical Society **6** (1955), no. 1, 111–116.

[51] X. Fang and K.-C. Toh, *Using a distributed SDP approach to solve simulated protein molecular conformation problems*, Distance Geometry, Springer, 2013, pp. 351–376.

[52] O. D. Faugeras and M. Hebert, *The representation, recognition, and locating of 3d objects*, International Journal of Robotics Research **5** (1986), no. 3, 27–52.

[53] M. P. Foster, C. A. McElroy, and C. D. Amero, *Solution NMR of large molecules and assemblies*, Biochemistry **46** (2007), no. 2, 331–340.

[54] D. Garber and E. Hazan, *Approximating semidefinite programs in sublinear time*, Advances in Neural Information Processing Systems, 2011, pp. 1080–1088.

[55] D. M. Gavrila, *The visual analysis of human movement: A survey*, Computer vision and image understanding **73** (1999), no. 1, 82–98.

[56] M. X. Goemans and D. P. Williamson, *Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming*, Journal of the ACM (JACM) **42** (1995), no. 6, 1115–1145.

[57] G. H. Golub and C. F. Van Loan, *Matrix Computations*, Johns Hopkins University Press, 1996.

[58] J. D. Gorman and A. O. Hero, *Lower bounds for parametric estimation with constraints*, Information Theory, IEEE Transactions on **36** (1990), no. 6, 1285–1301.

[59] S. Gortler, C. Gotsman, L. Liu, and D. Thurston, *On affine rigidity*, Journal of Computational Geometry **4** (2013), no. 1, 160–181.

[60] S. J. Gortler, A. D. Healy, and D. P. Thurston, *Characterizing generic global rigidity*, American Journal of Mathematics **132** (2010), no. 4, 897–939.

[61] S. J. Gortler and D. P. Thurston, *Characterizing the universal rigidity of generic frameworks*, arXiv preprint arXiv:1001.0172 (2009).

[62] C. Gotsman and Y. Koren, *Distributed graph layout for sensor networks*, Graph Drawing, Springer, 2004, pp. 273–284.

[63] J. C. Gower and G. B. Dijksterhuis, *Procrustes Problems*, vol. 3, Oxford University Press Oxford, 2004.

[64] M. Grant, S. Boyd, and Y. Ye, *CVX: Matlab software for disciplined convex programming*, 2008.

[65] A. M. Gronenborn, D. R. Filpula, N. Z. Essig, A. Achari, M. Whitlow, P. T. Wingfield, and G. M. Clore, *A novel, highly stable fold of the immunoglobulin binding domain of streptococcal protein G*, Science **253** (1991), no. 5020, 657–661.

[66] J. Groß, *Linear regression*, vol. 175, Springer Science & Business Media, 2003.

[67] P. Güntert, *Automated NMR structure calculation with CYANA*, Protein NMR Techniques, Springer, 2004, pp. 353–378.

[68] ———, *Automated structure determination from NMR spectra*, European Biophysics Journal **38** (2009), no. 2, 129–143.

[69] P. Güntert, C. Mumenthaler, and K. Wüthrich, *Torsion angle dynamics for NMR structure calculation with the new program DYANA*, Journal of molecular biology **273** (1997), no. 1, 283–298.

[70] E. Hartman, Z. Wang, Q. Zhang, K. Roy, G. Chanfreau, and J. Feigon, *Intrinsic dynamics of an extended hydrophobic core in the S. cerevisiae RNase iii dsrbd contributes to recognition of specific RNA binding sites*, Journal of molecular biology **425** (2013), no. 3, 546–562.

[71] T. F. Havel, I. D. Kuntz, and G. M. Crippen, *The combinatorial distance geometry method for the calculation of molecular conformation. i. a new approach to an old problem*, Journal of Theoretical Biology **104** (1983), no. 3, 359–381.

[72] C. Helmberg, F. Rendl, R. J. Vanderbei, and H. Wolkowicz, *An interior-point method for semidefinite programming*, SIAM Journal on Optimization **6** (1996), no. 2, 342–361.

[73] B. Hendrickson, *Conditions for unique graph realizations*, SIAM Journal on Computing **21** (1992), no. 1, 65–84.

[74] J. N. Higham, *Computing the polar decomposition - with applications*, SIAM Journal on Scientific and Statistical Computing **7** (1986), no. 4, 1160–1174.

[75] N.-D. Ho and P. Van Dooren, *On the pseudo-inverse of the Laplacian of a bipartite graph*, Applied Mathematics Letters **18** (2005), no. 8, 917–922.

[76] B. K. P. Horn, *Closed-form solution of absolute orientation using unit quaternions*, JOSA A **4** (1987), no. 4, 629–642.

[77] S. D. Howard, D. Cochran, W. Moran, and F. R. Cohen, *Estimation and registration on graphs*, arXiv:1010.2983 (2010).

[78] W. Hu and L. Wang, *Residual dipolar couplings: Measurements and applications to biomolecular studies*, Annual Reports on NMR Spectroscopy **58** (2006), 231–303.

[79] Qi-Xing Huang and Leonidas Guibas, *Consistent shape maps via semidefinite programming*, Eurographics Symposium on Geometry Processing, vol. 32, 2013, pp. 177–186.

[80] B. Jackson, *Notes on the rigidity of graphs*, Levico Conference Notes, vol. 4, Citeseer, 2007.

[81] A. Javanmard and A. Montanari, *Localization from incomplete noisy distance measurements*, Foundations of Computational Mathematics **13** (2013), no. 3, 297–345.

[82] M. Journee, F. Bach, P.-A. Absil, and R. Sepulchre, *Low-rank optimization on the cone of positive semidefinite matrices*, SIAM Journal on Optimization **20** (2010), no. 5, 2327–2351.

[83] Y-S Jung and M. Zweckstetter, *Mars-robust automatic backbone assignment of proteins*, Journal of biomolecular NMR **30** (2004), no. 1, 11–23.

[84] W. Kabsch, *A solution for the best rotation to relate two sets of vectors*, Acta Crystallographica Section A: Crystal Physics, Diffraction, Theoretical and General Crystallography **32** (1976), no. 5, 922–923.

[85] N. Karmarkar, *A new polynomial-time algorithm for linear programming*, Proceedings of the sixteenth annual ACM symposium on Theory of computing, ACM, 1984, pp. 302–311.

[86] J. Keeler, *Understanding NMR spectroscopy*, John Wiley & Sons, 2011.

[87] J. B. Keller, *Closest unitary, orthogonal and hermitian operators to a given operator*, Mathematics Magazine **48** (1975), no. 4, 192–197.

[88] L. G. Khachiyan, *Polynomial algorithms in linear programming*, USSR Computational Mathematics and Mathematical Physics **20** (1980), no. 1, 53–72.

[89] Y. Khoo, A. Singer, and D. Cowburn, *Bias correction in saupe tensor estimation*, to be submitted (2016).

[90] ———, *Integrating noe and rdc using semidefinite programming for protein structure determination*, arXiv preprint arXiv:1604.01504 (2016).

[91] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, *Optimization by simulated annealing*, Science **220** (1983), no. 4598, 671–680.

[92] G. Kontaxis, F. Delaglio, and A. Bax, *Molecular fragment replacement approach to protein structure determination by chemical shift and dipolar homology database mining*, Methods in Enzymology **394** (2005), 42–78.

[93] J. Kowalewski and L. Maler, *Nuclear spin relaxation in liquids: theory, experiments, and applications*, CRC Press, 2006.

[94] S. Krishnan, P. Y. Lee, J. B. Moore, and S. Venkatasubramanian, *Global registration of multiple 3d point sets via optimization-on-a-manifold*, Eurographics Symposium on Geometry Processing (2005), 187–197.

[95] L. D. Landau and E. M. Lifshitz, *Quantum mechanics: non-relativistic theory*, vol. 3, Elsevier, 2013.

[96] G. Lerman, M. McCoy, J. A. Tropp, and T. Zhang, *Robust computation of linear models, or how to find a needle in a haystack*, arXiv:1202.4044 (2012).

[97] N-H Leung and K-C Toh, *An SDP-based divide-and-conquer algorithm for large-scale noisy anchor-free graph realization*, SIAM Journal on Scientific Computing **31** (2009), no. 6, 4351–4372.

[98] R.-C. Li, *New perturbation bounds for the unitary polar factor*, SIAM Journal on Matrix Analysis and Applications **16** (1995), no. 1, 327–332.

[99] L. Liberti, C. Lavor, N. Maculan, and A. Mucherino, *Euclidean distance geometry and applications*, SIAM Review **56** (2014), no. 1, 3–69.

[100] R. S. Lipsitz and N. Tjandra, *Residual dipolar couplings in NMR structure analysis*, Annu. Rev. Biophys. Biomol. Struct. **33** (2004), 387–413.

[101] J. A. Losonczi, M. Andrec, M. W. Fischer, and J. H. Prestegard, *Order matrix analysis of residual dipolar couplings using singular value decomposition*, Journal of Magnetic Resonance **138** (1999), no. 2, 334–342.

[102] L. Lovász and M. D. Plummer, *Matching theory*, vol. 367, American Mathematical Soc., 2009.

[103] L. Lovász and A. Schrijver, *Cones of matrices and set-functions and 0-1 optimization*, SIAM Journal on Optimization **1** (1991), no. 2, 166–190.

[104] S. C. Lovell, J. M. Word, J. S. Richardson, and D. C. Richardson, *The penultimate rotamer library*, Proteins: Structure, Function, and Bioinformatics **40** (2000), no. 3, 389–408.

[105] J. Meiler, J. J. Prompers, W. Peti, C. Griesinger, and R. Brüschweiler, *Model-free approach to the dynamic interpretation of residual dipolar couplings in globular proteins*, Journal of the American Chemical Society **123** (2001), no. 25, 6098–6107.

[106] E. Meirovitch, D. Lee, K. F. Walter, and C. Griesinger, *Standard tensorial analysis of local ordering in proteins from residual dipolar couplings*, The Journal of Physical Chemistry B **116** (2012), no. 21, 6106–6117.

[107] L. Mirsky, *Symmetric gauge functions and unitarily invariant norms*, The Quarterly Journal of Mathematics **11** (1960), no. 1, 50–59.

[108] N. J. Mitra, N. Gelfand, H. Pottmann, and L. Guibas, *Registration of point cloud data from a geometric optimization perspective*, Eurographics Symposium on Geometry Processing, 2004, pp. 22–31.

[109] R. Mukhopadhyay, S. Irausquin, C. Schmidt, and H. Valafar, *DYNAFOLD: A dynamic programming approach to protein backbone structure determination from minimal sets of residual dipolar couplings*, Journal of bioinformatics and computational biology **12** (2014), no. 01, 1450002.

[110] A. Naor, O. Regev, and T. Vidick, *Efficient rounding for the noncommutative rothendieck inequality*, Proceedings of the 45th Annual ACM Symposium on Theory of Computing, 2013, pp. 71–80.

[111] A. Nemirovski, *Sums of random symmetric matrices and quadratic optimization under orthogonality constraints*, Mathematical Programming **109** (2007), no. 2-3, 283–317.

[112] Y. Nesterov, *Semidefinite relaxation and nonconvex quadratic optimization*, Optimization methods and software **9** (1998), no. 1-3, 141–160.

[113] D. Neuhaus, *Nuclear overhauser effect*, eMagRes (2000).

[114] R. Peng and M. L. Sichitiu, *Angle of arrival localization for wireless sensor networks*, Sensor and Ad Hoc Communications and Networks, 2006. SECON'06. 2006 3rd Annual IEEE Communications Society on, vol. 1, IEEE, 2006, pp. 374–382.

[115] H. Pottmann, Q.-X. Huang, Y.-L. Yang, and S.-M. Hu, *Geometry and convergence analysis of algorithms for registration of 3d shapes*, International Journal of Computer Vision **67** (2006), no. 3, 277–296.

[116] G. N. Ramachandran, C. T. Ramakrishnan, and V. Sasisekharan, *Stereochemistry of polypeptide chain configurations*, Journal of molecular biology **7** (1963), no. 1, 95–99.

[117] G. Ranjan, Z.-L. Zhang, and D. Boley, *Incremental computation of pseudo-inverse of Laplacian: Theory and applications*, arXiv:1304.2300 (2013).

[118] A. G. Redfield, *On the theory of relaxation processes*, IBM Journal of Research and Development **1** (1957), no. 1, 19–31.

[119] R. T. Rockafellar, *Lagrange multipliers and optimality*, SIAM review **35** (1993), no. 2, 183–238.

[120] ———, *Convex analysis*, Princeton university press, 2015.

[121] S. Rusinkiewicz and M. Levoy, *Efficient variants of the ICP algorithm*, Proceedings of the Third International Conference on 3d Digital Imaging and Modeling, 2001, pp. 145–152.

[122] T. M. Sabo, C. A. Smith, D. Ban, A. Mazur, D. Lee, and C. Griesinger, *Orium: Optimized rdc-based iterative and unified model-free analysis*, Journal of biomolecular NMR **58** (2014), no. 4, 287–301.

[123] R. Sanyal, F. Sottile, and B. Sturmfels, *Orbitopes*, Mathematika **57** (2011), no. 02, 275–314.

[124] J. Saunderson, P. A. Parrilo, and A. S. Willsky, *Semidefinite descriptions of the convex hull of rotation matrices*, SIAM Journal on Optimization **25** (2015), no. 3, 1314–1343.

[125] J. B. Saxe, *Embeddability of weighted graphs in k-space is strongly NP-hard*, Carnegie-Mellon University, Department of Computer Science, 1980.

[126] J. Schäfer and K. Strimmer, *A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics*, Statistical applications in genetics and molecular biology **4** (2005), no. 1.

[127] E. Schmidt and P. GuÌĹntert, *A new algorithm for reliable and general NMR resonance assignment*, Journal of the American Chemical Society **134** (2012), no. 30, 12817–12829.

[128] C. D. Schwieters, J. J. Kuszewski, N. Tjandra, and G. M. Clore, *The Xplor-NIH NMR molecular structure determination package*, Journal of Magnetic Resonance **160** (2003), no. 1, 65–73.

[129] G. C. Sharp, S. W. Lee, and D. K. Wehe, *Multiview registration of 3d scenes by minimizing error between coordinate frames*, Proceedings of the 7th European Conference on Computer Vision-Part II, Springer-Verlag, 2002, pp. 587–597.

[130] Y. Shen, F. Delaglio, G. Cornilescu, and A. Bax, *Talos+: a hybrid method for predicting protein backbone torsion angles from NMR chemical shifts*, Journal of biomolecular NMR **44** (2009), no. 4, 213–223.

[131] J. Shi and J. Malik, *Normalized cuts and image segmentation*, IEEE Transactions on Pattern Analysis and Machine Intelligence **22** (2000), no. 8, 888–905.

[132] A. Simonetto and G. Leus, *Distributed maximum likelihood sensor network localization*, Signal Processing, IEEE Transactions on **62** (2014), no. 6, 1424–1437.

[133] A. Singer, *A remark on global positioning from local distances*, Proceedings of the National Academy of Sciences **105** (2008), no. 28, 9507–9511.

[134] ———, *Angular synchronization by eigenvectors and semidefinite programming*, Applied and computational harmonic analysis **30** (2011), no. 1, 20–36.

[135] A. Singer and M. Cucuringu, *Uniqueness of low-rank matrix completion by rigidity theory*, SIAM Journal on Matrix Analysis and Applications **31(4)** (2010), 1621–1641.

[136] A. M.-C. So, *Moment inequalities for sums of random matrices and their applications in optimization*, Mathematical Programming **130** (2011), no. 1, 125–151.

[137] A.M.C. So and Y. Ye, *Theory of semidefinite programming for sensor network localization*, Mathematical Programming **109** (2007), no. 2-3, 367–384.

[138] D. A. Spielman and S.-H. Teng, *Nearly-linear time algorithms for graph partitioning, graph sparsification, and solving linear systems*, Proceedings of the Thirty-sixth Annual ACM Symposium on Theory of Computing, 2004, pp. 81–90.

[139] L. A. Stefanski and J. R. Cook, *Simulation-extrapolation: the measurement error jackknife*, Journal of the American Statistical Association **90** (1995), no. 432, 1247–1256.

[140] P. Stoica and B. C. Ng, *On the Cramér-Rao bound under parametric constraints*, Signal Processing Letters, IEEE **5** (1998), no. 7, 177–179.

[141] K.-C. Toh, M. J. Todd, and R. H. Tutuncu, *SDPT3 – A matlab software package for semidefinite programming, version 1.3*, Optimization Methods and Software **11** (1999), no. 1-4, 545–581.

[142] J. R. Tolman, *A novel approach to the retrieval of structural and dynamic information from residual dipolar couplings using several oriented media in biomolecular nmr spectroscopy*, Journal of the American Chemical Society **124** (2002), no. 40, 12020–12030.

[143] J. R. Tolman and K. Ruan, *NMR residual dipolar couplings as probes of biomolecular dynamics*, Chemical Reviews **106** (2006), no. 5, 1720–1736.

[144] J. W. Tukey, *Exploratory data analysis*, (1977).

[145] T. Tzeneva, *Global alignment of multiple 3d scans using eigevector synchronization*, Senior Thesis, Princeton University (supervised by S. Rusinkiewicz and A. Singer) (2011).

[146] L. Vandenberghe and S. Boyd, *Semidefinite programming*, SIAM review **38** (1996), no. 1, 49–95.

[147] S. Vijay-Kumar, C. E. Bugg, and W. J. Cook, *Structure of ubiquitin refined at 1.8 åresolution*, Journal of molecular biology **194** (1987), no. 3, 531–544.

[148] N. K. Vishnoi, *Lx=b*, Foundations and Trends in Theoretical Computer Science **8** (2012), no. 1-2, 1–141.

[149] I. Waldspurger, A. d'Aspremont, and S. Mallat, *Phase recovery, maxcut and complex semidefinite programming*, arXiv:1206.0102 (2012).

[150] L. Wang and B. R. Donald, *Exact solutions for internuclear vectors and backbone dihedral angles from NH residual dipolar couplings in two media, and their application in a systematic search algorithm for determining protein backbone structure*, Journal of Biomolecular NMR **29** (2004), no. 3, 223–242.

[151] L. Wang and A. Singer, *Exact and stable recovery of rotations for robust synchronization*, Information and Inference (2013), iat005.

[152] L. Wang and A. Singer, *Exact and stable recovery of rotations for robust synchronization*, Information and Inference: A Journal of the IMA, accepted for publication (2013).

[153] Z. Wang, S. Zheng, Y. Ye, and S. Boyd, *Further relaxations of the semidefinite programming approach to sensor network localization*, SIAM Journal on Optimization **19** (2008), no. 2, 655–673.

[154] J. M. Ward and N. R. Skrynnikov, *Very large residual dipolar couplings from deuterated ubiquitin*, Journal of biomolecular NMR **54** (2012), no. 1, 53–67.

[155] K. Q. Weinberger and L. K. Saul, *An introduction to nonlinear dimensionality reduction by maximum variance unfolding*, AAAI, vol. 6, 2006, pp. 1683–1686.

[156] Z. Wen, D. Goldfarb, S. Ma, and K. Scheinberg, *Block coordinate descent methods for semidefinite programming*, Handbook on Semidefinite, Conic and Polynomial Optimization, Springer, 2012, pp. 533–564.

[157] Z. Wen, D. Goldfarb, and W. Yin, *Alternating direction augmented lagrangian methods for semidefinite programming*, Mathematical Programming Computation **2** (2010), no. 3-4, 203–230.

[158] W. Whiteley, *Counting out to the flexibility of molecules*, Physical Biology **2** (2005), no. 4, S116.

[159] J. A. Williams and M. Bennamoun, *Simultaneous registration of multiple point sets using orthonormal matrices*, Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, vol. 6, 2000, pp. 2199–2202.

[160] M. P. Williamson and C. J. Craven, *Automated protein structure calculation from NMR data*, Journal of biomolecular NMR **43** (2009), no. 3, 131–143.

[161] L. N. Wirz and J. R. Allison, *Fitting alignment tensor components to experimental RDCs, CSAs and RQCs*, Journal of Biomolecular NMR (2015), 1–5.

[162] H. Wolkowicz and M. F. Anjos, *Semidefinite programming for discrete optimization and matrix completion problems*, Discrete Applied Mathematics **123** (2002), no. 1, 513–577.

[163] H. Wolkowicz, R. Saigal, and L. Vandenberghe, *Handbook of Semidefinite Programming: Theory, Algorithms, and Applications*, vol. 27, Kluwer Academic Pub, 2000.

[164] Z. Wu, F. Delaglio, K. Wyatt, G. Wistow, and A. Bax, *Solution structure of γs-crystallin by molecular fragment replacement NMR*, Protein science **14** (2005), no. 12, 3101–3114.

[165] Y. Xu, Y. Zheng, J-S Fan, and D. Yang, *A new strategy for structure determination of large proteins in solution without deuteration*, Nature methods **3** (2006), no. 11, 931–937.

[166] A. Yershova, C. Tripathy, P. Zhou, and B. R. Donald, *Algorithms and analytic solutions using sparse residual dipolar couplings for high-resolution automated protein backbone structure determination by NMR*, Algorithmic Foundations of Robotics IX, Springer, 2011, pp. 355–372.

[167] J. Zeng, J. Boyles, C. Tripathy, L. Wang, A. Yan, P. Zhou, and B. R. Donald, *High-resolution protein structure determination starting with a global fold calculated from exact solutions to the RDC equations*, Journal of biomolecular NMR **45** (2009), no. 3, 265–281.

[168] H. Zha and Z. Zhang, *Spectral properties of the alignment matrices in manifold learning*, SIAM review **51** (2009), no. 3, 545–566.

[169] L. Zhang, L. Liu, C. Gotsman, and S. Gortler, *An As-Rigid-As-Possible approach to sensor network localization*, ACM Transactions on Sensor Networks **6** (2010), no. 4, 35.

[170] L. Zhi-Quan, M. Wing-Kin, A.M.-C. So, Y. Ye, and S. Zhang, *Semidefinite relaxation of quadratic optimization problems*, IEEE Signal Processing Magazine **27** (2010), no. 3, 20–34.

[171] H. Zhou, *1D and 2D NOESY Comparison*, `http://nmr.chem.ucsb.edu/protocols/noesy_compare.html`.

[172] M. Zweckstetter, *NMR: prediction of molecular alignment from structure using the PALES software*, Nature protocols **3** (2008), no. 4, 679–690.

[173] M. Zweckstetter and A. Bax, *Evaluation of uncertainty in alignment tensors obtained from dipolar couplings*, Journal of Biomolecular NMR **23** (2002), no. 2, 127–137.