PAPER

Non-unique games over compact groups and orientation estimation in cryo-EM

View the article online for updates and enhancements.

Non-unique games over compact groups and orientation estimation in cryo-EM

Afonso S Bandeira[1], Yutong Chen[2,5], Roy R Lederman[3] and Amit Singer[4]

[1] Department of Mathematics, ETH Zürich, Ramistrasse 101, 8092 Zürich, Switzerland
[2] Systems Trading, Tudor Investment Corporation, New York, NY United States of America
[3] Department of Statistics and Data Science, Yale University, 24 Hillhouse Avenue, New Haven, CT 06511 United States of America
[4] Department of Mathematics, Program in Applied and Computational Mathematics, Princeton University, Fine Hall, Washington Road, Princeton, NJ 08544 United States of America

E-mail: bandeira@math.ethz.ch, yutongc@gmail.com, roy.lederman@yale.edu and amits@math.princeton.edu

Abstract

Let $\mathcal{G}$ be a compact group and let $f_{ij} \in C(\mathcal{G})$. We define the *non-unique games* (NUG) problem as finding $g_1, \ldots, g_n \in \mathcal{G}$ to minimize $\sum_{i,j=1}^{n} f_{ij} \left(g_i g_j^{-1} \right)$. We introduce a convex relaxation of the NUG problem to a *semidefinite program* (SDP) by taking the Fourier transform of f_{ij} over $\mathcal{G}$. The NUG framework can be seen as a generalization of the *little Grothendieck* problem over the orthogonal group and the *unique games* problem and includes many practically relevant problems, such as the *maximum likelihood estimator* to registering bandlimited functions over the unit sphere in *d*-dimensions and orientation estimation of noisy cryo-electron microscopy (cryo-EM) projection images. We implement an SDP solver for the NUG cryo-EM problem using the alternating direction method of multipliers (ADMM). Numerical study with synthetic datasets indicate that while our ADMM solver is slower than existing methods, it can estimate the rotations more accurately, especially at low *signal-to-noise ratio* (SNR).

(Some figures may appear in colour only in the online journal)

---

[5] Author to whom any correspondence should be addressed.

1. Introduction

We consider problems of the following form:

$$
\begin{aligned}
&\underset{g_1,\ldots,g_n}{\text{minimize}} \quad \sum_{i,j=1}^{n} f_{ij}\left(g_i g_j^{-1}\right) \\
&\text{subject to} \quad g_i \in \mathcal{G},
\end{aligned}
\tag{1.1}
$$

where $\mathcal{G}$ is a compact group and $f_{ij} : \mathcal{G} \to \mathbb{R}$ are suitable functions. We will refer to such problems as a *non-unique game* (NUG) problem over $\mathcal{G}$.

Note that the solution to the NUG problem is not unique. If $g_1, \ldots, g_n$ is a solution to (1.1), then so is $g_1 g, \ldots, g_n g$ for any $g \in \mathcal{G}$. That is, we can solve (1.1) up to a global shift $g \in \mathcal{G}$.

In many inverse problems, the goal is to estimate multiple group elements from information about group offsets, and can be formulated as (1.1). A simple example is angular synchronization [1], where one is tasked with estimating angles $\{\theta_i\}_i$ from information about their offsets $\theta_i - \theta_j \bmod 2\pi$. The problem of estimating the angles can then be formulated as an optimization problem depending on the offsets, and thus be written in the form of (1.1). In this case, $\mathcal{G} \cong SO(2)$.

One of the simplest instances of (1.1) is the `Max-Cut` problem, where the objective is to partition the vertices of a graph as to maximize the number of edges (the *cut*) between the two sets. In this case, $\mathcal{G} \cong \mathbb{Z}_2$, the group of two elements $\{\pm 1\}$, and f_{ij} is zero if (i,j) is not an edge of the graph and

$$
f_{ij}(1) = 0 \quad \text{and} \quad f_{ij}(-1) = -1,
$$

if (i,j) is an edge. In fact, we take a semidefinite programming based approach toward (1.1) that is inspired by—and can be seen as a generalization of—the semidefinite relaxation for the `Max-Cut` problem by Goemans and Williamson [2].

Another important source of inspiration is the semidefinite relaxation of `Max-2-Lin`$(\mathbb{Z}_L)$, proposed in [3], for the *unique games* problem, a central problem in theoretical computer science [4, 5]. Given integers n and L, an unique-games instance is a system of linear equations over $\mathbb{Z}_L$ on n variables $\{x_i\}_{i=1}^{n}$. Each equation constraints the difference of two variables. More precisely, for each (i,j) in a subset of the pairs, we associate a constraint

$$
x_i - x_j = b_{ij} \mod L.
$$

The objective is then to find $\{x_i\}_{i=1}^{n}$ in $\mathbb{Z}_L$ that satisfy as many equations as possible. This can be easily described within our framework by taking, for each constraint,

$$
f_{ij}(g) = -\delta_{g \equiv b_{ij}},
$$

and $f_{ij} = 0$ for pairs not corresponding to constraints. The term 'unique' derives from the fact that the constraints have this special structure where the offset can only take one value to satisfy the constraint, and all other values have the same score. This motivated our choice of nomenclature for the framework treated in this paper. The semidefinite relaxation for the unique games problem proposed in [3] was investigated in [6] in the context of the signal alignment problem, where the f_{ij} are not forced to have a special structure (but $\mathcal{G} \cong \mathbb{Z}_L$). The NUG framework presented in this paper can be seen as a generalization of the approach in [6]

to other compact groups $\mathcal{G}$. We emphasize that, unlike [6] that was limited to the case of a finite cyclic group, here we consider compact groups that are possibly infinite and non-commutative.

Besides the signal alignment problem treated in [6] the semidefinite relaxation to the NUG problem we develop generalizes with other effective relaxations. When $\mathcal{G} \cong \mathbb{Z}_2$ it coincides with the semidefinite relaxations for Max-Cut [2], the little Grothendieck problem over $\mathbb{Z}_2$ [7, 8], recovery in the stochastic block model [9, 10], and synchronization over $\mathbb{Z}_2$ [10–12]. When $\mathcal{G} \cong SO(2)$ and the functions f_{ij} are linear with respect to the representation $\rho_1 : SO(2) \to \mathbb{C}$ given by $\rho_1(\theta) = \mathrm{e}^{\mathrm{i}\theta}$, it coincides with the semidefinite relaxation for angular synchronization [1]. Similarly, when $\mathcal{G} \cong O(d)$ and the functions are linear with respect to the natural d-dimensional representation, then the NUG problem essentially coincides with the little Grothendieck problem over the orthogonal group [13, 14]. Other examples include the shape matching problem in computer graphics for which $\mathcal{G}$ is the permutation group (see [15, 16]). In addition, it has been shown in [17] that the formulation of NUG and the algorithms presented in this paper can be extended to simultaneous alignment and classification of a mixture of different signals.

1.1. Orientation estimation in cryo-electron microscopy

A particularly important application of this framework is the orientation estimation problem in cryo-electron microscopy [18].

Cryo-EM is a technique used to determine the three-dimensional structure of biological macromolecules. The molecules are rapidly frozen in a thin layer of ice and imaged with an electron microscope, which gives noisy two-dimensional projections. One of the main difficulties with this imaging process is that these molecules are imaged at different unknown orientations in the sheet of ice and each molecule can only be imaged once (due to the destructive nature of the imaging process). More precisely, each measurement consists of a tomographic projection of a rotated (by an unknown rotation) copy of the molecule. The task is then to reconstruct the molecule density from many such noisy measurements. Although in principle it is possible to reconstruct the three-dimensional density directly from the noisy images without estimation of the rotations [19], or by treating rotations as nuisance parameters [20, 21] here we consider the problem of estimating the rotations directly from the noisy images. In section 2, we describe how this problem can be formulated in the form (1.1).

2. Multireference Alignment

In classical linear inverse problems, one is tasked with recovering an unknown element $x \in \mathcal{X}$ from a noisy measurement of the form $\mathcal{P}(x) + \epsilon$, where ϵ represents the measurement error and $\mathcal{P}$ is a linear observation operator. There are, however, many problems where an additional difficulty is present; one class of such problems includes non-linear inverse problems in which an unknown transformation acts on x prior to the linear measurement. Specifically, let $\mathcal{X}$ be a vector space and $\mathcal{G}$ be a group acting on $\mathcal{X}$. Suppose we have n measurements of the form

$$y_i = \mathcal{P}(g_i \circ x) + \epsilon_i, \quad i = 1, \ldots, n \tag{2.1}$$

where

- x is a fixed but unknown element of $\mathcal{X}$,
- $g_1, \ldots, g_n$ are unknown elements of $\mathcal{G}$,
- $\circ$ is the action of $\mathcal{G}$ on $\mathcal{X}$,
- $\mathcal{P} : \mathcal{X} \to Y$ is a linear operator,

- Y is the (finite-dimensional) measurement space,
- ϵ_i's are independent noise terms.

If the g_i's were known, then the task of recovering x would reduce to a classical linear inverse problem, for which many effective techniques exist. While in many situations it is possible to estimate x directly without estimating $g_1, \ldots, g_n$, or by treating these as nuisance parameters, here we focus on the problem of estimating the group elements $g_1, \ldots, g_n$.

There are several common approaches for inverse problems of the form (2.1). One is motivated by the observation that estimating x knowing the g_i's and estimating the g_i's knowing x are both considerably easier tasks. This suggests an alternating minimization approach where each estimation is updated iteratively. Besides a lack of theoretical guarantees, convergence may also depend on the initial guess. Another approach, which we refer to as pairwise comparisons [1], consists in determining, from pairs of observations (y_i, y_j), the most likely value for $g_i g_j^{-1}$. Although the problem of estimating the g_i's from these pairwise guesses is fairly well-understood [1, 22, 23] enjoying efficient algorithms and performance guarantees, this method suffers from loss of information as not all of the information of the problem is captured in this most likely value for $g_i g_j^{-1}$ and thus this approach tends to fail at low signal-to-noise-ratio.

In contrast, the *maximum likelihood estimator* (MLE) leverages all information. Assuming that the ϵ_i's are i.i.d. Gaussian, the MLE for the observation model (2.1) is given by the following optimization problem:

$$
\begin{aligned}
\underset{g_1, \ldots, g_n, x}{\text{minimize}} \quad & \sum_{i=1}^{n} \| y_i - \mathcal{P}(g_i \circ x) \|_2^2 \\
\text{subject to} \quad & g_i \in \mathcal{G} \\
& x \in \mathcal{X}
\end{aligned}
\tag{2.2}
$$

We refer to (2.2) as the *multireference alignment* (MRA) problem. Let us denote the ground truth signal and group elements by x^0 and $g_1^0, \ldots, g_n^0$; the solution to the optimization problem by x^* and $g_1^*, \ldots, g_n^*$, which we will also refer to as x^{MLE} and $g_1^{\text{MLE}}, \ldots, g_n^{\text{MLE}}$. Unfortunately, the exponentially large search space and nonconvex nature of (2.2) often render it computationally intractable. However, for several problems of interest, we formulate (2.2) as an instance of an NUG for which we develop computationally tractable approximations.

Notice that although MLE typically enjoys several theoretical properties, their underlying technical conditions do not hold in this case. Specifically, the number of parameters to be estimated is not fixed but rather grows indefinitely with the sample size n: for each sample y_i there is a group element g_i that needs to be estimated. As a result, the MLE may not be consistent in this case. In other words, even in the limit $n \to \infty$ the estimator x^{MLE} may not converge to the ground truth x^0. Similarly, the estimated group elements will not converge to their true values. A different version of MLE, not considered in this paper, in which the group elements are treated as nuisance parameters and are marginalized would enjoy the nice theoretical properties.

2.1. Registration of signals on the sphere

Consider the problem of estimating a bandlimited signal on the circle $x : S^1 \to \mathbb{C}$ from noisy rotated discrete sampled copies of it. In this problem, $\mathcal{X} = \text{span}\{e^{ik\theta}\}_{k=-t}^{t}$ is the space of
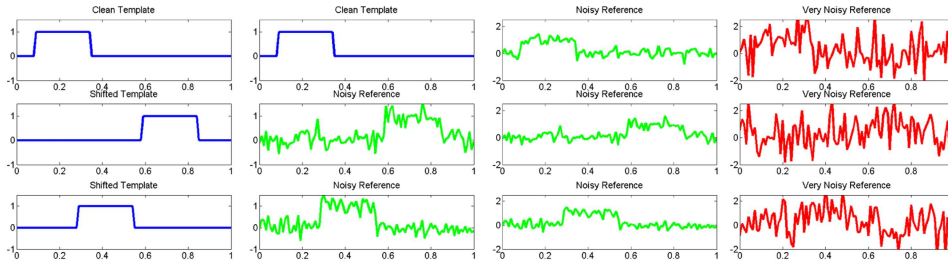
Figure 1. Illustration of the registration problem in S^1. The first column consists of a noiseless signal at three different shifts, the second column represents an instance for which the template x is known and *matched filtering* is effective to estimate the shifts. However, in the examples we are interested in the template is unknown (last two columns) rendering the problem of estimating the shifts significantly harder.

bandlimited functions up to degree t on S^1, $\mathcal{G} = SO(2)$ and the group action is

$$g \circ x = \sum_{k=-t}^{t} \alpha_k e^{ik(\theta - \theta_g)},$$

where $x = \sum_{k=-t}^{t} \alpha_k e^{ik\theta}$ and we identified $g \in SO(2)$ with $\theta_g \in [0, 2\pi]$.

The measurements are of the form

$$y_i := \mathcal{P}(g_i \circ x) + \epsilon_i, \quad i = 1, \ldots, n$$

where

- $x \in \mathcal{X}$,
- $g_i \in SO(2)$,
- $\mathcal{P} : \mathcal{X} \to \mathbb{C}^L$ samples the function at L equally spaced points in S^1,
- $\epsilon_i \sim \mathcal{N}(0, \sigma^2 I_{L \times L})$ ($i = 1, \ldots, n$) are independent Gaussians.

Our objective is to estimate $g_1, \ldots, g_n$ and x. Since estimating x knowing the group elements g_i is considerably easier, we will focus on estimating $g_1, \ldots, g_n$. As shown below, this will essentially reduce to the problem of aligning (or registering) the observations $y_1, \ldots, y_n$.

In absence of noise, the problem of finding the g_i's is trivial (cf first column of figure 1). With noise, if x is known (as it is in some applications), then the problem of determining the g_i's can be solved by *matched filtering* (cf second column of figure 1). However, x is unknown in general. This, together with the high levels of noise, render the problem significantly more difficult (cf last two columns of figure 1).

We now define the problem of registration in d-dimensions in general. $\mathcal{X} = \text{span}\{p_k\}_{k \in \mathcal{A}_t}$ is the space of bandlimited functions up to degree t on S^d where the p_k's are orthonormal polynomials on S^d, $\mathcal{A}_t$ indexes all p_k up to degree t and $\mathcal{G} = SO(d+1)$.

The measurements are of the form

$$y_i := \mathcal{P}(g_i \circ x) + \epsilon_i, \quad i = 1, \ldots, n \tag{2.3}$$

where

- $x \in \mathcal{X}$,
- $g_i \in SO(d+1)$,
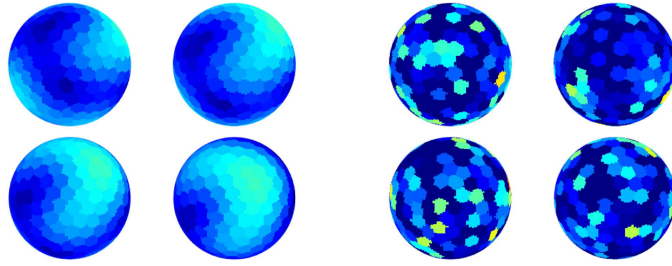
Figure 2. An illustration of registration in two-dimensions. The left four spheres provide examples of clean signals y_i and the right four spheres are of noisy observations. Note that the images are generated using a quantization of the sphere.

- $\mathcal{P} : \mathcal{X} \to \mathbb{C}^L$ samples the function on L points in S^d,
- $\epsilon_i \sim \mathcal{N}(0, \sigma^2 I_{L \times L})$ ($i = 1, \ldots, n$) are independent Gaussians.

Again, our objective is to estimate $g_1, \ldots, g_n$ and x. We would like the sampling operator $\mathcal{P}$ to be 'uniform'. One possible sampling scheme is *spherical designs* surveyed in [24]. An illustration of signals on a sphere, sampled at such points, is provided in figure 2.

The MRA solution for registration in d-dimensions is given by

$$
\begin{aligned}
\underset{g_1, \ldots, g_n, x}{\text{minimize}} \quad & \sum_{i=1}^{n} \| y_i - \mathcal{P}(g_i \circ x) \|_2^2 \\
\text{subject to} \quad & g_i \in SO(d+1) \\
& x \in \mathcal{X}
\end{aligned}
\tag{2.4}
$$

We now remove x from (2.4). Let $\mathcal{Q} : \mathbb{C}^L \to \mathcal{X}$ be the adjoint of $\mathcal{P}$. $\mathcal{Q}$ is also an approximate inverse of $\mathcal{P}$ (up to normalization), because points are sampled from a t-design which has the property of exactly integrating polynomials on the sphere. Then, $\| y_i - \mathcal{P}(g_i \circ x) \|_2^2 \approx \| \mathcal{Q}(y_i) - g_i \circ x \|_2^2$ (up to normalization), and the approximation error decreases as L increases. Since g_i preserves the norm $\| \cdot \|_2$, it follows that (2.4) is equivalent to

$$
\begin{aligned}
\underset{g_1, \ldots, g_n, x}{\text{minimize}} \quad & \sum_{i=1}^{n} \| g_i^{-1} \circ \mathcal{Q}(y_i) - x \|_2^2 \\
\text{subject to} \quad & g_i \in SO(d+1) \\
& x \in \mathcal{X}.
\end{aligned}
\tag{2.5}
$$

Since the minimizer x with fixed g_i's is the average, (2.5) is equivalent to

$$
\begin{aligned}
\underset{g_1, \ldots, g_n}{\text{minimize}} \quad & \sum_{i,j=1}^{n} \| g_i^{-1} \circ \mathcal{Q}(y_i) - g_j^{-1} \circ \mathcal{Q}(y_j) \|_2^2 \\
\text{subject to} \quad & g_i \in SO(d+1).
\end{aligned}
\tag{2.6}
$$
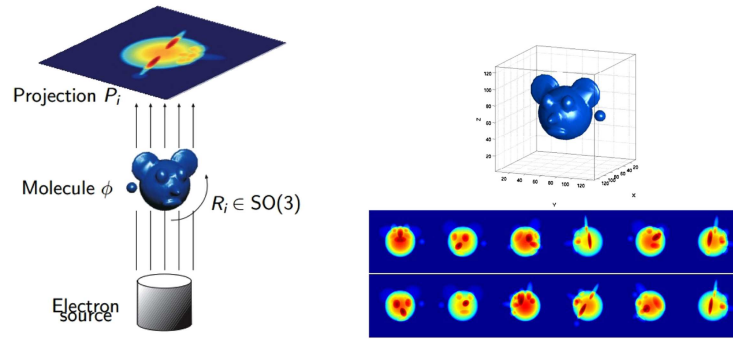
Figure 3. Illustration of the cryo-EM imaging process: a molecule is imaged after being frozen at a random (unknown) rotation and a tomographic two-dimensional projection is captured. Given a number of tomographic projections taken at unknown rotations, we are interested in determining such rotations with the objective of reconstructing the molecule density. Images courtesy of Singer and Shkolnisky [27].

Figure 4. Sample images from the *E. coli* 50S ribosomal subunit, generously provided by Dr. Fred Sigworth at the Yale Medical School.

Since g_i preserves $\|\cdot\|_2$ norm, then (2.6) is equivalent to

$$
\begin{aligned}
&\underset{g_1,\ldots,g_n}{\text{minimize}} \quad \sum_{i,j=1}^{n} \| \mathcal{Q}(y_i) - g_i g_j^{-1} \circ \mathcal{Q}(y_j) \|_2^2 \\
&\text{subject to} \quad g_i \in SO(d+1).
\end{aligned}
\tag{2.7}
$$

In summary, (2.4) can be approximated by (2.7), which is an instance of (1.1).
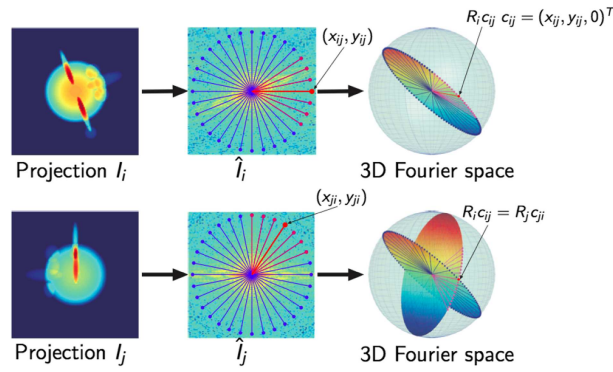
2.2. Orientation estimation in cryo-EM

The task here is to reconstruct the molecule density from many noisy tomographic projection images (see the right column of figure 3 for an idealized density and measurement dataset). We assume the molecule does not have any non-trivial point group symmetry. The linear inverse problem of recovering the molecule density given the rotations fits in the framework of classical computerized tomography for which effective methods exist. Thus, we focus on the non-linear inverse problem of estimating the unknown rotations and the underlying density.

An added difficulty is the high level of noise in the images. In fact, it is already non-trivial to distinguish whether a molecule is present in an image or if the image consists only of noise (see figure 4 for a subset of an experimental dataset). On the other hand, these datasets consist of many projection images which renders reconstruction possible.

We formulate the problem of orientation estimation in cryo-EM. Let $\mathcal{X}$ be the space of bandlimited functions that are also essentially compactly supported in $\mathbb{R}^3$ and $\mathcal{G} = SO(3)$.

Figure 5. An illustration of the use of the Fourier slice theorem and the common lines approach to the orientation estimation problem in cryo-EM. Image courtesy of Singer and Shkolnisky [27].

For perfectly centered images, and ignoring the effect of the microscope's contrast transfer function, the measurements are of the form

$$I_i(x, y) := \mathcal{P}(g_i \circ \phi) + \epsilon_i, \quad i = 1, \dots, n \tag{2.8}$$

- $\phi \in \mathcal{X}$,
- $g_i \in SO(3)$,
- $\mathcal{P}(\phi)$ samples $\int_{-\infty}^{\infty} \phi(x, y, z) \, dz$ ($\mathcal{P}$ is called the discrete x-ray transform),
- ϵ_i's are i.i.d Gaussians representing noise.

Our objective is to find $g_1, \dots, g_n$ and ϕ.

The operator $\mathcal{P}$ in the orientation estimation problem is different than in the registration problem. Specifically, $\mathcal{P}$ is a composition of tomographic projection and sampling. To write the objective function for the orientation estimation problem, we will use the *Fourier slice theorem* [26].

The *Fourier slice theorem* states that the two-dimensional Fourier transform of a tomographic projection of a molecular density ϕ coincides with the restriction to a plane normal to the projection direction, a slice, of the three-dimensional Fourier transform of the density ϕ. See figure 5.

Let $\hat{I}_i(r, \theta)$ be the Fourier transform of I_i in polar coordinates. We identify $\hat{I}_i$ and $\hat{I}_j$ with the xy-plane in $\mathbb{R}^3$, and apply g_i^{-1} and g_j^{-1} to $\hat{I}_i$ and $\hat{I}_j$, respectively. Then, the directions of the lines of intersection on $\hat{I}_i$ and $\hat{I}_j$ are given, respectively, by unit vectors

$$c_{ij}\left(g_i g_j^{-1}\right) = \frac{g_i(g_i^{-1} \cdot \vec{e}_3 \times g_j^{-1} \cdot \vec{e}_3)}{\|g_i(g_i^{-1} \cdot \vec{e}_3 \times g_j^{-1} \cdot \vec{e}_3)\|_2} = \frac{\vec{e}_3 \times g_i g_j^{-1} \cdot \vec{e}_3}{\|\vec{e}_3 \times g_i g_j^{-1} \cdot \vec{e}_3\|_2}, \tag{2.9}$$

$$c_{ji}\left(g_i g_j^{-1}\right) = \frac{g_j(g_i^{-1} \cdot \vec{e}_3 \times g_j^{-1} \cdot \vec{e}_3)}{\|g_j(g_i^{-1} \cdot \vec{e}_3 \times g_j^{-1} \cdot \vec{e}_3)\|_2} = \frac{\left(g_i g_j^{-1}\right)^{-1} \cdot \vec{e}_3 \times \vec{e}_3}{\|\left(g_i g_j^{-1}\right)^{-1} \cdot \vec{e}_3 \times \vec{e}_3\|_2}. \tag{2.10}$$

where $\vec{e}_3 := (0, 0, 1)^T$. See [27] for details.

Since the noiseless images should agree on their common lines, we consider the following MRA-like cost function:

$$
\underset{g_1,\dots,g_n}{\text{minimize}} \quad \sum_{i,j=1}^{n} \|\hat{I}_i\left(\cdot, c_{ij}\left(g_i g_j^{-1}\right)\right) - \hat{I}_j\left(\cdot, c_{ji}\left(g_i g_j^{-1}\right)\right)\|_2^2
$$

$$
\text{subject to} \quad g_i \in SO(3),
$$

(2.11)

where, with a minor abuse of notation, we identify the vector c_{ij} with the angle θ_{ij} of a common line in the Fourier transform of an image $\hat{I}_i$. Equation (2.11) is an instance of (1.1). Note that we could also use the L_1 norm or a weighted L_2 norm in the cost function.

Note that for $n = 2$ images, there is always a degree of freedom along the line of intersection. In other words, we cannot recover the true orientation between $\hat{I}_1$ and $\hat{I}_2$. However, for $n \geqslant 3$, this degree of freedom is eliminated. It is also worth mentioning several important references in the context of angular reconstruction [28, 29]. In general, the measurement system suffers from a handedness ambiguity on the reconstruction (see, for example [27]), this will be discussed in detail later in the paper.

3. Linearization via Fourier expansion

Let us consider the objective function in the general form

$$
\sum_{i,j=1}^{n} f_{ij}\left(g_i g_j^{-1}\right).
$$

(3.1)

Note that each f_{ij} in (3.1) can be nonlinear and nonconvex. However, since $\mathcal{G}$ is compact (and since $f_{ij} \in C(\mathcal{G})$), we can expand, each f_{ij} in *Fourier series*. More precisely, given the unitary irreducible representations $\{\rho_k\}$ of $\mathcal{G}$, we can write

$$
f_{ij}\left(g_i g_j^{-1}\right) = \sum_{k=0}^{\infty} d_k \text{tr}\left[\hat{f}_{ij}(k)\rho_k\left(g_i g_j^{-1}\right)\right]
$$

$$
= \sum_{k=0}^{\infty} d_k \text{tr}\left[\hat{f}_{ij}(k)\rho_k(g_i)\rho_k^*(g_j)\right],
$$

(3.2)

where $\hat{f}_{ij}(k)$ are the *Fourier coefficients* of f_{ij} and can be computed from f_{ij} via the *Fourier transform*

$$
\hat{f}_{ij}(k) := \int_{\mathcal{G}} f_{ij}(g)\rho_k(g^{-1})\mathrm{d}g
$$

$$
= \int_{\mathcal{G}} f_{ij}(g)\rho_k^*(g)\mathrm{d}g.
$$

(3.3)

Above, $\mathrm{d}g$ denotes the *Haar measure* on $\mathcal{G}$ and d_k the dimension of the representation ρ_k. See [30] for an introduction to the representations of compact groups.

We express the objective function (3.1) as

$$
\sum_{i,j=1}^{n} f_{ij}\left(g_i g_j^{-1}\right) = \sum_{i,j=1}^{n} \sum_{k=0}^{\infty} d_k \mathrm{tr}\left[\hat{f}_{ij}(k)\rho_k(g_i)\rho_k^*(g_j)\right]
$$

$$
= \sum_{k=0}^{\infty} \sum_{i,j=1}^{n} d_k \mathrm{tr}\left[\hat{f}_{ij}(k)\rho_k(g_i)\rho_k^*(g_j)\right],
$$

which is linear in $\rho_k(g_i)\rho_k^*(g_j)$. This motivates writing (1.1) as linear optimization over the variables

$$
X^{(k)} := \begin{bmatrix} \rho_k(g_1) \\ \vdots \\ \rho_k(g_n) \end{bmatrix} \begin{bmatrix} \rho_k(g_1) \\ \vdots \\ \rho_k(g_n) \end{bmatrix}^*.
$$

In other words,

$$
\sum_{i,j=1}^{n} f_{ij}\left(g_i g_j^{-1}\right) = \sum_{k=0}^{\infty} \mathrm{tr}\left[C^{(k)} X^{(k)}\right],
$$

where the coefficient matrices are given by

$$
C^{(k)} := d_k \begin{bmatrix} \hat{f}_{11}(k) & \hat{f}_{21}(k) & \cdots & \hat{f}_{n1}(k) \\ \hat{f}_{12}(k) & \hat{f}_{22}(k) & \cdots & \hat{f}_{n2}(k) \\ \vdots & \vdots & \ddots & \vdots \\ \hat{f}_{1n}(k) & \hat{f}_{2n}(k) & \cdots & \hat{f}_{nn}(k) \end{bmatrix}.
$$

We refer to the $d_k \times d_k$ block of $X^{(k)}$ corresponding to $\rho_k(g_i)\rho_k^*(g_j) = \rho_k(g_i g_j^{-1})$ as $X_{ij}^{(k)}$. We now turn our attention to constraints on the variables $\left\{X^{(k)}\right\}_{k=0}^{\infty}$. It is easy to see that:

$$
X^{(k)} \succeq 0, \quad \forall_k \tag{3.4}
$$

$$
X_{ii}^{(k)} = I_{d_k \times d_k}, \quad \forall_{k,i}, \tag{3.5}
$$

$$
\mathrm{rank}\left[X^{(k)}\right] = d_k, \quad \forall_k, \tag{3.6}
$$

$$
X_{ij}^{(k)} \in \mathrm{Im}(\rho_k), \quad \forall_{k,i,j}. \tag{3.7}
$$

Constraints (3.4), (3.5) and (3.6) ensure $X^{(k)}$ is of the form

$$
X^{(k)} = \begin{bmatrix} X_1^{(k)} \\ X_2^{(k)} \\ \vdots \\ X_n^{(k)} \end{bmatrix} \begin{bmatrix} X_1^{(k)} \\ X_2^{(k)} \\ \vdots \\ X_n^{(k)} \end{bmatrix}^*,
$$

for some $X_i^{(k)}$ unitary $d_k \times d_k$ matrices. The constraint (3.7) attempts to ensure that $X_i^{(k)}$ is in the image of the representation of $\mathcal{G}$. Notably, none of these constraints ensures that, for different

values of k, $X_{ij}^{(k)}$ correspond to the same group element $g_i g_j^{-1}$. Adding such a constraint would yield

$$
\begin{aligned}
\underset{X^{(k)}}{\text{minimize}} \quad & \sum_{k=0}^{\infty} \text{tr}\left[C^{(k)}X^{(k)}\right] \\
\text{subject to} \quad & X^{(k)} \succeq 0 \\
& X_{ii}^{(k)} = I_{d_k \times d_k} \\
& \text{rank}\left[X^{(k)}\right] = d_k \\
& X_{ij}^{(k)} = \rho_k(g_i g_j^{-1}),
\end{aligned}
\tag{3.8}
$$

where g_i and g_j are elements of $\mathcal{G}$.

Unfortunately, both the rank constraint and the last constraint in (3.8) are, in general, non-convex. We will relax (3.8) by dropping the rank requirement and replacing the last constraint by positivity constraints that couple different $X^{(k)}$'s. We achieve this by considering the *Dirac delta* funcion on $\mathcal{G}$. Notice that the Dirac delta funcion $\delta(g)$ on the identity $e \in \mathcal{G}$ can be expanded as

$$
\begin{aligned}
\delta(g) &= \sum_{k=0}^{\infty} d_k \text{tr}\left[\hat{\delta}(k)\rho_k(g)\right] \\
&= \sum_{k=0}^{\infty} d_k \text{tr}\left[\left(\int_{\mathcal{G}} \delta(h)\rho_k^*(h)dh\right)\rho_k(g)\right] \\
&= \sum_{k=0}^{\infty} d_k \text{tr}\left[\rho_k(g)\right].
\end{aligned}
$$

If we replace g with $g^{-1}\left(g_i g_j^{-1}\right)$, then we get

$$
\begin{aligned}
\delta(g^{-1}g_i g_j^{-1}) &= \sum_{k=0}^{\infty} d_k \text{tr}\left[\rho_k(g^{-1})\rho_k\left(g_i g_j^{-1}\right)\right] \\
&= \sum_{k=0}^{\infty} d_k \text{tr}\left[\rho_k^*(g)X_{ij}^{(k)}\right].
\end{aligned}
$$

To arrive at a convex program, we consider the following convex constraints, that form a natural convex relaxation for Dirac deltas,

$$
\sum_{k=0}^{\infty} d_k \text{tr}\left[\rho_k^*(g)X_{ij}^{(k)}\right] \geqslant 0 \quad \forall g \in \mathcal{G},
\tag{3.9}
$$

$$
\int_{\mathcal{G}}\left(\sum_{k=0}^{\infty} d_k \text{tr}\left[\rho_k^*(g)X_{ij}^{(k)}\right]\right)dg = 1.
\tag{3.10}
$$

This suggests relaxing (3.8) to

$$
\begin{aligned}
\underset{X^{(k)}}{\text{minimize}} \quad & \sum_{k=0}^{\infty} \text{tr}\left[C^{(k)}X^{(k)}\right] \\
\text{subject to} \quad & X^{(k)} \succeq 0 \\
& X_{ii}^{(k)} = I_{d_k \times d_k} \\
& \sum_{k=0}^{\infty} d_k \text{tr}\left[\rho_k^*(g)X_{ij}^{(k)}\right] \geqslant 0 \quad \forall g \in \mathcal{G} \\
& \int_{\mathcal{G}} \left(\sum_{k=0}^{\infty} d_k \text{tr}\left[\rho_k^*(g)X_{ij}^{(k)}\right]\right) \mathrm{d}g = 1.
\end{aligned}
\tag{3.11}
$$

For a nontrivial irreducible representation ρ_k, we have $\int_{\mathcal{G}} \rho_k(g)\mathrm{d}g = 0$. This means that the integral constraint in (3.11) is equivalent to the constraint

$$
X_{ij}^{(0)} = 1, \quad \forall_{i,j}.
$$

Thus, we focus on the optimization problem

$$
\begin{aligned}
\underset{X^{(k)}}{\text{minimize}} \quad & \sum_{k=0}^{\infty} \text{tr}\left[C^{(k)}X^{(k)}\right] \\
\text{subject to} \quad & X^{(k)} \succeq 0 \\
& X_{ii}^{(k)} = I_{d_k \times d_k} \\
& \sum_{k=0}^{\infty} d_k \text{tr}\left[\rho_k^*(g)X_{ij}^{(k)}\right] \geqslant 0 \quad \forall g \in \mathcal{G} \\
& X_{ij}^{(0)} = 1.
\end{aligned}
\tag{3.12}
$$

When $\mathcal{G}$ is a finite group it has only a finite number of irreducible representations. This means that (3.12) is a semidefinite program and can be solved, to arbitrary precision, in polynomial time [31]. In fact, when $\mathcal{G} \cong \mathbb{Z}_L$, a suitable change of basis shows that (3.12) is equivalent to the semidefinite programming relaxation proposed in [6] for the *signal alignment problem*.

Unfortunately, many of the applications of interest involve infinite groups. This creates two obstacles to solving (3.12). One is due to the infinite sum in the objective function and the other due to the infinite number of positivity constraints. In the next section, we address these two obstacles for the groups $SO(2)$ and $SO(3)$.

4. Finite truncations for *SO*(2) and *SO*(3) via *Fejér kernels*

The objective of this section is to replace (3.12) by an optimization problem depending only in finitely many variables $X^{(k)}$. The objective function in (3.12) is converted from an infinite sum to a finite sum by truncating at degree t. That is, we fix a t and set $C^{(k)} = 0$ for $k > t$. This

consists of truncating the Fourier series of $\sum_{i,j=1}^{n} f_{ij}\left(g_i g_j^{-1}\right)$. Unfortunately, constraint (3.9) given by

$$\sum_{k=0}^{\infty} d_k \mathrm{tr}\left[\rho_k^*(g) X_{ij}^{(k)}\right] \geqslant 0 \quad \forall g \in \mathcal{G},$$

still involves infinitely many variables $X_{ij}^{(k)}$ and consists of infinitely many linear constraints.

We now address this issue for the groups $SO(2)$ and $SO(3)$.

4.1. Truncation for SO(2)

Since we truncated the objective function at degree t, it is then natural to truncate the infinite sum in constraint (3.9) also at t. If we truncated below t, then some variables (such as $X^{(t)}$) are not constrained; and if we truncated above t, then some variables (such as $X^{(t+1)}$) do not affect the cost function. The irreducible representations of $SO(2)$ are $\{e^{ik\theta}\}$, and $d_k = 1$ for all k. Let us identify $g \in SO(2)$ with $\theta_g \in [0, 2\pi]$. That straightforward truncation corresponds to approximating the Dirac delta with

$$\delta(g) \approx \sum_{k=-t}^{t} e^{ik\theta_g}.$$

This approximation is known as the *Dirichlet kernel*, which we denote as

$$D_t(\theta) := \sum_{k=-t}^{t} e^{ik\theta}.$$

However, the Dirichlet kernel does not inherit all the desirable properties of the delta function. In fact, $D_t(\theta)$ is negative for some values of θ.

Instead, we use the *Fejér kernel*, which is a non-negative kernel, to approximate the Dirac delta. The *Fejér kernel* is defined as

$$F_t(\theta) := \frac{1}{t}\sum_{k=0}^{t-1} D_k = \sum_{k=-t}^{t} \left(1 - \frac{|k|}{t}\right) e^{ik\theta},$$

which is the first-order *Cesàro mean* of the Dirichlet kernel.

This motivates us to replace constraint (3.9) with

$$\sum_{k=-t}^{t} \left(1 - \frac{|k|}{t}\right) e^{-ik\theta} X_{ij}^{(k)} \geqslant 0 \quad \forall \theta \in [0, 2\pi],$$

where, for $k > 0$, $X_{ij}^{(-k)}$ denotes $\left[X_{ij}^{(k)}\right]^*$.

This suggests considering

$$\underset{X^{(k)}}{\text{minimize}} \quad \sum_{k=0}^{t} \text{tr}\left[C^{(k)}X^{(k)}\right]$$

$$\text{subject to} \quad X^{(k)} \succeq 0$$

$$X_{ii}^{(k)} = I_{d_k \times d_k}$$

$$\sum_{k=-t}^{t} \left(1 - \frac{|k|}{t}\right) e^{-ik\theta} X_{ij}^{(k)} \geqslant 0 \quad \forall \theta \in [0, 2\pi]$$

$$X_{ij}^{(0)} = 1,$$

which only depends on the variables $X_{ij}^{(k)}$ for $k = 0, \ldots, t$.

Unfortunately, the condition that the trigonometric polynomial $\sum_{k=-t}^{t} \left(1 - \frac{|k|}{t}\right) e^{-ik\theta} X_{ij}^{(k)}$ is always non-negative, still involves an infinite number of linear inequalities. Interestingly, due to the Fejér–Riesz factorization theorem (see [32]), this condition can be replaced by an equivalent condition involving a positive semidefinite matrix—it turns out that every nonnegative trigonometric polynomial is a square, meaning that the so called *sum-of-squares relaxation* [33, 34] is exact. However, while such a formulation would still be an SDP and thus solvable, up to arbitrary precision, in polynomial time, it would involve a positive semidefinite variable for every pair (i, j), rendering it computationally challenging. For this reason we relax the non-negativity constraint by asking that $\sum_{k=-t}^{t} \left(1 - \frac{|k|}{t}\right) e^{-ik\theta} X_{ij}^{(k)}$ is non-negative in a finite set $\Omega_t \in SO(2)$. This yields the following optimization problem:

$$\underset{X^{(k)}}{\text{minimize}} \quad \sum_{k=0}^{t} \text{tr}\left[C^{(k)}X^{(k)}\right]$$

$$\text{subject to} \quad X^{(k)} \succeq 0$$

$$X_{ii}^{(k)} = I_{d_k \times d_k} \tag{4.1}$$

$$\sum_{k=-t}^{t} \left(1 - \frac{|k|}{t}\right) e^{-ik\theta} X_{ij}^{(k)} \geqslant 0 \quad \forall \theta \in \Omega_t$$

$$X_{ij}^{(0)} = 1.$$

4.2. Truncation for SO(3)

The irreducible representations of *SO*(3) are the *Wigner-D matrices* $\{W^{(k)}(\alpha, \beta, \gamma)\}$, and $d_k = 2k + 1$. See [35] for an introduction to *Wigner-D matrices*. Let us associate $g \in SO(3)$ with *Euler* (*Z–Y–Z*) *angle* $(\alpha, \beta, \gamma) \in [0, 2\pi] \times [0, \pi] \times [0, 2\pi]$. A straightforward truncation yields the approximation

$$\delta(g) \approx \sum_{k=0}^{t} (2k + 1) \text{tr}\left[W^{(k)}(\alpha, \beta, \gamma)\right].$$

Observe that the operator tr is invariant under conjugation. Then $W^{(k)}$ can be decomposed as

$$W^{(k)}(\alpha, \beta, \gamma) = R\Lambda^{(k)}(\theta)R^*$$

with an R such that

$$\Lambda^{(k)}(\theta) = \begin{bmatrix} e^{-ik\theta} & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & & \ddots & \\ & & & & e^{ik\theta} \end{bmatrix}.$$

We can think of R as a change of basis and $\Lambda^{(k)}(\theta)$ as a rotation from $SO(2)$ under the basis R. It follows that

$$\operatorname{tr}\left[W^{(k)}(\alpha, \beta, \gamma)\right] = \operatorname{tr}\left[\Lambda^{(k)}(\theta)\right] = \sum_{m=-k}^{k} e^{im\theta} = D_k(\theta).$$

The relationship between θ and α, β, γ is

$$\theta = 2\arccos\left[\cos\left(\frac{\beta}{2}\right)\cos\left(\frac{\alpha+\gamma}{2}\right)\right].$$

This relationship can be obtained by directly evaluating $\operatorname{tr}\left[W^{(1)}(\alpha, \beta, \gamma)\right]$ using the *Wigner-D matrix* $w^{(1)}$:

$$\operatorname{tr}\left[W^{(1)}(\alpha, \beta, \gamma)\right] = \sum_{m=-1}^{1} W_{m,m}^{(1)}(\alpha, \beta, \gamma)$$

$$= \sum_{m=-1}^{1} e^{-im(\alpha+\gamma)} w_{m,m}^{(1)}(\beta)$$

$$= \cos(\beta)(1 + \cos(\alpha+\gamma)) + \cos(\alpha+\gamma).$$

See [35] also for an introduction to *Wigner-D matrix*. This straightforward truncation at t yields

$$\delta(g) \approx \sum_{k=0}^{t} (2k+1)D_k(\theta),$$

which, again, inherits the undesirable property that this approximation can be negative for some θ. Recall that we circumvented this property in the one-dimension case by taking the first-order Cesàro mean of the Dirichlet kernel. In the two-dimension case, we will need the second-order Cesàro mean. Notice that

$$D_k(\theta) = \frac{\sin\left[(2k+1)\frac{\theta}{2}\right]}{\sin\left(\frac{\theta}{2}\right)}.$$

Fejér proved that [36]

$$\sum_{k=0}^{t} \frac{(3)_{t-k}}{(t-k)!}\left(k+\frac{1}{2}\right)\sin\left[(2k+1)\frac{\theta}{2}\right] \geqslant 0, \quad 0 \leqslant \theta \leqslant \pi$$

where $\frac{(3)_{t-k}}{(t-k)!} = \frac{1}{2}(t - k + 2)(t - k + 1)$. It follows that

$$\sum_{k=0}^{t} \frac{(3)_{t-k}}{(t-k)!} \left(k + \frac{1}{2}\right) D_k(\theta)$$

$$= \sum_{k=0}^{t} \frac{(3)_{t-k}}{(t-k)!} \left(k + \frac{1}{2}\right) \frac{\sin\left[(2k+1)\frac{\theta}{2}\right]}{\sin\left(\frac{\theta}{2}\right)} \geqslant 0, \quad -\pi \leqslant 0 \leqslant \pi.$$

Let us define

$$F_t(g) = F_t(\alpha, \beta, \gamma) := \sum_{k=0}^{t} \frac{(3)_{t-k}}{(t-k)!} \left(k + \frac{1}{2}\right) \frac{\sin\left[(2k+1)\frac{\theta_g}{2}\right]}{\sin\left(\frac{\theta_g}{2}\right)}$$

where $\theta_g = 2 \arccos \left[\cos\left(\frac{\beta}{2}\right) \cos\left(\frac{\alpha+\gamma}{2}\right)\right]$.

We replace constraint (3.9) with

$$F_t(\alpha, \beta, \gamma) \geqslant 0 \quad \forall(\alpha, \beta, \gamma) \in [0, 2\pi] \times [0, \pi] \times [0, 2\pi].$$

Secondly, we discretize the group $SO(3)$ to obtain a finite number of constraints. We consider a suitable finite subset $\Omega_t \subset SO(3)$. In our implementation, we use a *Hopf fibration* [37] to discretize $SO(3)$. The quotient space $SO(3)/SO(2)$ is equivalent to S^2. We take a uniform discretization of $S^1 \equiv SO(2)$ and a spherical design [24] of S^2. It is possible to find a spherical design on S^2 with $\mathcal{O}(r^2)$ points [24]. By [37], we use $\mathcal{O}(r)$ points to discretize $SO(2)$. The size of our $SO(3)$ discretization is $\mathcal{O}(r^3)$. So, we have to enforce $\mathcal{O}(r^3)$ inequality constraints. The choice of r is up to the user to strike a balance between computational speed and accuracy. We can then relax the non-negativity constraint yielding the following semidefinite program[6]:

$$\underset{X^{(k)}}{\text{minimize}} \sum_{k=0}^{t} \text{tr}\left[C^{(k)} X^{(k)}\right]$$

$$\text{subject to} X^{(k)} \succeq 0$$

$$X_{ii}^{(k)} = I_{d_k \times d_k} \tag{4.2}$$

$$\sum_{k=0}^{t} \frac{(3)_{t-k}}{(t-k)!} \left(k + \frac{1}{2}\right) \text{tr}\left[\left(W^{(k)}(\alpha, \beta, \gamma)\right)^* X_{ij}^{(k)}\right] \geqslant 0 \quad \forall(\alpha, \beta, \gamma) \in \Omega_t$$

$$X_{ij}^{(0)} = 1.$$

4.3. An additional constraint on $X^{(1)}$.

In this section, we discuss an additional constraint on $X^{(1)}$, which uses properties of quaternions to constrain each block $X_{ij}^{(1)}$ of $X^{(1)}$ in the convex hull of $SO(3)$ more directly.

We consider the standard rotation matrix R and the unit quaternion $q = q_r + q_i i + q_j j + q_k k$ which represent the same rotation as the block $X_{ij}^{(1)}$ (whose representation is associated with

---

[6] Similarly to $SO(2)$, it is possible that the non-negativity constraint may be replaced by an SDP or sums-of-squares constraint [38].

spherical harmonics), and we consider the outer product Q_{ij} of the unit quaternion q

$$Q_{ij} = q^T q. \tag{4.3}$$

The standard rotation matrix R is related to the block $X_{ij}^{(1)}$ by the formula

$$R^* = M^* \left(T^* X_{ij}^{(1)} T \right) M,$$

where,

$$T := \begin{bmatrix} -\dfrac{i}{\sqrt{2}} & 0 & \dfrac{1}{\sqrt{2}} \\ 0 & 1 & 0 \\ -\dfrac{i}{\sqrt{2}} & 0 & -\dfrac{1}{\sqrt{2}} \end{bmatrix},$$

and

$$M := \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -1 & 0 & 0 \end{bmatrix}.$$

Indeed, one can verify that

$$T^* X_{ij}^{(1)} T = \begin{bmatrix} R[22] & R[32] & -R[12] \\ R[23] & R[33] & -R[13] \\ -R[21] & -R[31] & R[11] \end{bmatrix},$$

where,

$$R = \begin{bmatrix} R[11] & R[12] & R[13] \\ R[21] & R[22] & R[23] \\ R[31] & R[32] & R[33] \end{bmatrix},$$

and M is used simply to rearrange the elements of $T^* X_{ij}^{(1)} T$.

Next, R is mapped to Q_{ij} by Rodrigues' rotation formula (see [39]):

$$Q_{ij} = \mathcal{R}(R) := \frac{1}{4} \begin{bmatrix} 1+R[11]+R[22]+R[33] & R[32]-R[23] & R[13]-R[31] & R[21]-R[12] \\ R[32]-R[23] & 1+R[11]-R[22]-R[33] & R[21]+R[12] & R[31]+R[13] \\ R[13]-R[31] & R[21]+R[12] & 1+R[22]-R[11]-R[33] & R[23]+R[32] \\ R[21]-R[12] & R[31]+R[13] & R[23]+R[32] & 1+R[33]-R[11]-R[22] \end{bmatrix}.$$

In summary, the mapping from $X_{ij}^{(1)}$ to Q_{ij}, which we denote by $\mathcal{A}_{E_q}$, is given by the formula:

$$Q_{ij} = \mathcal{A}_{E_q} \left(X_{ij}^{(1)} \right) = \mathcal{R}(M^* T^* X_{ij}^{(1)} TM).$$

For each Q_{ij}, we wish to impose the constraints implied by equation (4.3):

$$Q_{ij} \succeq 0, \quad \text{tr}[Q_{ij}] = 1, \quad \text{rank}[Q_{ij}] = 1.$$

But since the constraint rank$[Q_{ij}] = 1$ is not convex, we will drop it, giving us the following SDP in place of (4.2):

$$
\begin{aligned}
\underset{X^{(k)}}{\text{minimize}} \quad & \sum_{k=0}^{t} \text{tr}\left[C^{(k)}X^{(k)}\right] \\
\text{subject to} \quad & X^{(k)} \succeq 0 \\
& X_{ii}^{(k)} = I_{d_k \times d_k} \\
& \sum_{k=0}^{t} \frac{(3)_{t-k}}{(t-k)!}\left(k + \frac{1}{2}\right) \text{tr}\left[\left(W^{(k)}(\alpha, \beta, \gamma)\right)^* X_{ij}^{(k)}\right] \geqslant 0 \quad \forall(\alpha, \beta, \gamma) \in \Omega_t \\
& X_{ij}^{(0)} = 1 \\
& Q_{ij} = \mathcal{A}_{E_q}\left(X_{ij}^{(1)}\right), \quad Q_{ij} \succeq 0, \quad \text{tr}[Q_{ij}] = 1.
\end{aligned}
\tag{4.4}
$$

5. Applications

In this section, we consider the application of (4.1) to the problem of registration over the unit circle and the application of (4.4) to registration over the unit sphere and orientation estimation in cryo-EM. To solve the SDP for each problem, the only parameters we need to determine are the coefficient matrices $C^{(k)}$ and the truncation parameter t. The calculations of $C^{(k)}$ are detailed in the following pages. As for t, we experimented over a range of values for each problem, and chose a value that balances computational time with accuracy in the estimated rotations. The SDP outputs the relative transformations $g_{ij} = g_i g_j^{-1}$, while we need $g_1, \ldots, g_n$. For each problem, we describe a *rounding* procedure to recover $g_1, \ldots, g_n$ from the g_{ij}'s.

5.1. Registration in one-dimension

Recall that $\mathcal{X}$ is the space of bandlimited functions up to degree t on S^1. That is, for $x \in \mathcal{X}$, we can express

$$
x(\omega) = \sum_{l=-t}^{t} \alpha_l e^{il\omega}.
$$

Again, the irreducible representations of $SO(2)$ are $\{e^{ik\theta}\}$, and $d_k = 1$ for all k. Let us identify $g \in SO(2)$ with $\theta_g \in [0, 2\pi]$, then

$$
g \cdot x(\omega) = \sum_{l=-t}^{t} e^{il\theta_g} \alpha_l e^{il\omega}.
$$

Let $\mathcal{P}$ sample the underlying signal x at $L = 2t + 1$ distinct points. This way, we can determine all the α_l's associated with x.

Since $y_i = \mathcal{P}(g_i \cdot x) + \epsilon_i$, for the adjoint $\mathcal{Q}$, we have

$$
\mathcal{Q}(y_i)(\omega) = \sum_{l=-t}^{t} \alpha_l^{(i)} e^{il\omega}.
$$

Let us identify $g_i g_j^{-1}$ with $\theta_{ij} \in [0, 2\pi]$. Then, we can express f_{ij} in terms of $\alpha_l^{(i)}$, $\alpha_l^{(j)}$ and θ_{ij}:

$$
f_{ij}\left(g_i g_j^{-1}\right) = \|\mathcal{Q}(y_i) - g_i g_j^{-1} \circ \mathcal{Q}(y_j)\|_2^2
$$

$$
= \int_{S^1} \left| \sum_{l=-t}^{t} \left(\alpha_l^{(i)} - \alpha_l^{(j)} e^{il\theta_{ij}} \right) e^{il\omega} \right|^2 d\omega
$$

$$
= \sum_{l=-t}^{t} \left| \alpha_l^{(i)} - \alpha_l^{(j)} e^{il\theta_{ij}} \right|^2.
$$

The Fourier coefficients of f_{ij} are

$$
\hat{f}_{ij}(k) = \int_0^{2\pi} \sum_{l=-t}^{t} \left| \alpha_l^{(i)} - \alpha_l^{(j)} e^{il\theta_{ij}} \right|^2 e^{-ik\theta_{ij}} d\theta_{ij}
$$

$$
= \int_0^{2\pi} \sum_{l=-t}^{t} \left(|\alpha_l^{(i)}|^2 e^{-ik\theta_{ij}} + |\alpha_l^{(j)}|^2 e^{-ik\theta_{ij}} \right.
$$

$$
\left. - \alpha_l^{(i)} \overline{\alpha_l^{(j)}} e^{i(-k-l)\theta_{ij}} - \overline{\alpha_l^{(i)}} \alpha_l^{(j)} e^{i(-k+l)\theta_{ij}} \right) d\theta_{ij}
$$

$$
= 2\pi \begin{cases} \sum_{l=-t}^{t} \left(|\alpha_l^{(i)}|^2 + |\alpha_l^{(j)}|^2 \right) - \alpha_0^{(i)} \overline{\alpha_0^{(j)}} - \overline{\alpha_0^{(i)}} \alpha_0^{(j)} & , \quad k = 0 \\[2ex] -\alpha_{-k}^{(i)} \overline{\alpha_{-k}^{(j)}} - \overline{\alpha_k^{(i)}} \alpha_k^{(j)} & , \quad k \neq 0 \end{cases}
$$

Note that we re-indexed the coefficients $\hat{f}_{ij}(k) \leftarrow \hat{f}_{ij}(k - (t+1))$.

5.1.1. Rounding. (4.1) gives us the $X^{(k)}$'s. From the $X^{(k)}$'s, we want to extract each $\theta_i \in [0, 2\pi]$ (up to a global transformation). Let us consider $X^{(1)}$. We want $X^{(1)}$ to be of the form

$$
\begin{bmatrix} e^{i\theta_1} \\ e^{i\theta_2} \\ \vdots \\ e^{i\theta_n} \end{bmatrix} \begin{bmatrix} e^{i\theta_1} \\ e^{i\theta_2} \\ \vdots \\ e^{i\theta_n} \end{bmatrix}^*.
$$

Although $X^{(1)}$ is not guaranteed to be rank1, we will simply take the top eigenvector of $X^{(1)}$ as our estimate of $e^{i\theta_1}, \ldots, e^{i\theta_n}$. And from $e^{i\theta_1}, \ldots, e^{i\theta_n}$, we can recover $\theta_1, \ldots, \theta_n$. See [1] for the reasoning behind this approach. In practice, we find using the top eigenvector of $X^{(1)}$ is a sufficient estimate of $\theta_1, \ldots, \theta_n$. We do not need to run additional comparisons against $X^{(2)}, \ldots, X^{(t)}$.

5.2. Registration in two-dimension

Recall that $\mathcal{X}$ is the space of bandlimited functions up to degree t on S^2. That is, for $x \in \mathcal{X}$, we can express

$$
x(\omega) = \sum_{l=0}^{t} \sum_{m=-l}^{l} \alpha_{l,m} Y_{l,m}(\omega),
$$

where $\{Y_{lm}\}$ are the spherical harmonics. Again, the irreducible representations of $SO(3)$ are the Wigner D-matrices $\{W^{(k)}(\alpha, \beta, \gamma)\}$, and $d_k = 2k + 1$. Let us associate $g \in SO(3)$ with Euler $(Z-Y-Z)$ angle $(\alpha, \beta, \gamma) \in [0, 2\pi] \times [0, \pi] \times [0, 2\pi]$, then

$$g \cdot x(\omega) = \sum_{l=0}^{t} \sum_{m,m'=-l}^{l} W_{m,m'}^{(l)}(\alpha, \beta, \gamma)\alpha_{l,m'} Y_{l,m}(\omega).$$

Let $\mathcal{P}$ sample the underlying signal x at $L = (t + 1)^2$ points. This way, we can determine all the α_{lm}'s associated with x.

Again, for the adjoint $\mathcal{Q}$ we have

$$\mathcal{Q}(y_i)(\omega) = \sum_{l=0}^{t} \sum_{m=-l}^{l} \alpha_{l,m}^{(i)} Y_{l,m}(\omega).$$

Let us identify $g_i g_j^{-1} \in SO(3)$ with Euler $(Z-Y-Z)$ angle $(\alpha_{ij}, \beta_{ij}, \gamma_{ij}) \in [0, 2\pi] \times [0, 2\pi]$. Then, we can express f_{ij} in terms of $\alpha_{l,m}^{(i)}$, $\alpha_{l,m}^{(j)}$ and $(\alpha_{ij}, \beta_{ij}, \gamma_{ij})$:

$$
\begin{aligned}
f_{ij}\left(g_i g_j^{-1}\right) &= \|\mathcal{Q}(y_i) - g_i g_j^{-1} \circ \mathcal{Q}(y_j)\|_2^2 \\
&= \sum_{l=0}^{t} \sum_{m=-l}^{l} \left(|\alpha_{l,m}^{(i)}|^2 + |\alpha_{l,m}^{(j)}|^2\right) \\
&\quad - \sum_{l=0}^{t} \sum_{m,m'=-l}^{l} \alpha_{l,m}^{(i)} \overline{W_{m,m'}^{(l)}}(\alpha_{ij}, \beta_{ij}, \gamma_{ij})\overline{\alpha_{l,m'}^{(j)}} \\
&\quad - \sum_{l=0}^{t} \sum_{m,m'=-l}^{l} \overline{\alpha_{l,m}^{(i)}} W_{m,m'}^{(l)}(\alpha_{ij}, \beta_{ij}, \gamma_{ij})\alpha_{l,m'}^{(j)}.
\end{aligned}
$$

The Fourier coefficients are given by

$$
\begin{aligned}
\hat{f}_{ij}(k) &= \int_{SO(3)} f_{ij}(g)\left(W^{(k)}(\alpha, \beta, \gamma)\right)^* \mathrm{d}g \\
&= \frac{8\pi^2}{2k+1}
\begin{cases}
\sum_{l=0}^{t} \sum_{m=-l}^{l} \left(|\alpha_{lm}^{(i)}|^2 + |\alpha_{lm}^{(j)}|^2\right) - \alpha_{00}^{(i)}\overline{\alpha_{00}^{(j)}} - \overline{\alpha_{00}^{(i)}}\alpha_{00}^{(j)} & , \quad k = 0 \\[2mm]
\left(-(-1)^{m-m'}\alpha_{k,-m'}^{(i)}\overline{\alpha_{k,-m}^{(j)}} - \overline{\alpha_{k,m'}^{(i)}}\alpha_{k,m}^{(j)}\right)_{m,m'=-k}^{k} & , \quad k \neq 0
\end{cases}
\end{aligned}
$$

Here, we used the orthogonality relationship

$$\int_{SO(3)} \overline{W_{m_1,m_2}^{(k)}}(\alpha, \beta, \gamma) W_{m_1',m_2'}^{(k')}(\alpha, \beta, \gamma)\mathrm{d}g = \frac{8\pi^2}{2k+1}\delta_{k,k'}\delta_{m_1,m_1'}\delta_{m_2,m_2'},$$

and the property

$$W_{m,m'}^{(k)}(\alpha, \beta, \gamma) = (-1)^{m-m'}\overline{W_{-m,-m'}^{(k)}}(\alpha, \beta, \gamma).$$

5.2.1. Rounding. Again, (4.4) gives us the $X^{(k)}$'s. From the $X^{(k)}$'s, we want to extract each $(\alpha, \beta, \gamma) \in [0, 2\pi] \times [0, \pi] \times [0, 2\pi]$ (up to a global transformation). Let us consider $X^{(1)}$. We want $X^{(1)}$ to be of the form

$$
\begin{bmatrix} W^{(1)}(\alpha_1, \beta_1, \gamma_1) \\ W^{(1)}(\alpha_2, \beta_2, \gamma_2) \\ \vdots \\ W^{(1)}(\alpha_n, \beta_n, \gamma_n) \end{bmatrix} \begin{bmatrix} W^{(1)}(\alpha_1, \beta_1, \gamma_1) \\ W^{(1)}(\alpha_2, \beta_2, \gamma_2) \\ \vdots \\ W^{(1)}(\alpha_n, \beta_n, \gamma_n) \end{bmatrix}^{*},
$$

where each $W^{(1)}(\alpha_i, \beta_i, \gamma_i)$ is a 3×3 matrix. Similarly, $X^{(1)}$ is not guaranteed to be rank3, but we will simply take the top three eigenvector of $X^{(1)}$ as our estimate of $W^{(1)}(\alpha_1, \beta_1, \gamma_1), \ldots, W^{(1)}(\alpha_n, \beta_n, \gamma_n)$. And from $W^{(1)}(\alpha_1, \beta_1, \gamma_1), \ldots, W^{(1)}(\alpha_n, \beta_n, \gamma_n)$, we can recover $(\alpha_1, \beta_1, \gamma_1), \ldots, (\alpha_n, \beta_n, \gamma_n)$.

5.3. Orientation estimation in cryo-EM

We refer to [40] to expand the objective function. We emphasize that the theory holds for arbitrary basis on the space containing the $\hat{I}_i$'s. We choose to construct the $C^{(k)}$'s using coefficients and parameters from the Fourier–Bessel expansion. Projection $\hat{I}_i$ can be expanded via Fourier–Bessel series as

$$
\hat{I}_i(r, \theta) = \sum_{k=-\infty}^{\infty} \sum_{q=1}^{\infty} \alpha_{kq}^{(i)} \psi_{kq}^{(c)}(r, \theta),
$$

where

$$
\psi_{kq}^{(c)}(r, \theta) = \begin{cases} N_{kq} J_k \left(R_{kq} \dfrac{r}{c} \right) e^{ik\theta} & , \quad r \leqslant c, \\ 0 & , \quad r > c. \end{cases}
$$

The parameters above are defined as follows:

- c is the radius of the disc containing the support of $\hat{I}_i$,
- J_k is the Bessel function of integer order k,
- R_{kq} is the qth root of J_k,
- $N_{kq} = \frac{1}{c\sqrt{\pi}|J_{k+1}(R_{kq})|}$ is a normalization factor.

To avoid aliasing, we truncate the Fourier-Bessel expansion as follows.

$$
\hat{I}_i(r, \theta) \approx \sum_{k=-k_{\max}}^{k_{\max}} \sum_{q=1}^{p_k} \alpha_{kq}^{(i)} \psi_{kq}^{(c)}(r, \theta).
$$

See [40] for a discussion on $k_{\max}$ and p_k. For the purpose of this section, let us assume we have $\{\alpha_{kq}^{(i)} : -k_{\max} \leqslant k \leqslant k_{\max}, 1 \leqslant q \leqslant p_k\}$ for each $\hat{I}_i$. (These can be computed from the Cartesian grid sampled images.)

We shall determine the relationship between $\hat{I}_i(r, \theta_i)$ and $\hat{I}_j(r, \theta_j)$, and the lines of intersection between $g_i^{-1} \cdot \hat{I}_i$ and $g_j^{-1} \cdot \hat{I}_j$ embedded in $\mathbb{R}^3$. Recall from (2.9) and (2.10) that the directions of the lines of intersection between $g_i^{-1} \cdot \hat{I}_i$ and $g_j^{-1} \cdot \hat{I}_j$ are given, respectively, by unit vectors

$$c_{ij}\left(g_i g_j^{-1}\right) = \frac{\vec{e}_3 \times g_i g_j^{-1} \cdot \vec{e}_3}{\|\vec{e}_3 \times g_i g_j^{-1} \cdot \vec{e}_3\|_2},$$

$$c_{ji}\left(g_i g_j^{-1}\right) = \frac{\left(g_i g_j^{-1}\right)^{-1} \cdot \vec{e}_3 \times \vec{e}_3}{\|\left(g_i g_j^{-1}\right)^{-1} \cdot \vec{e}_3 \times \vec{e}_3\|_2}.$$

Let us associate $g_i g_j^{-1} \in SO(3)$ with Euler (Z–Y–Z) angle $(\alpha_{ij}, \beta_{ij}, \gamma_{ij}) \in [0, 2\pi] \times [0, \pi] \times [0, 2\pi]$. Then

$$\vec{e}_3 \times g_i g_j^{-1} \cdot \vec{e}_3 = \begin{bmatrix} \sin \gamma_{ij} \sin \beta_{ij} \\ -\cos \gamma_{ij} \sin \beta_{ij} \\ 0 \end{bmatrix},$$

$$\left(g_i g_j^{-1}\right)^{-1} \cdot \vec{e}_3 \times \vec{e}_3 = \begin{bmatrix} -\sin \alpha_{ij} \sin \beta_{ij} \\ -\cos \alpha_{ij} \sin \beta_{ij} \\ 0 \end{bmatrix},$$

under the rotation matrix $R_Z(\gamma_{ij}) R_Y(\beta_{ij}) R_Z(\alpha_{ij})$. The directions of the lines of intersection in $\hat{I}_i$ and $\hat{I}_j$ under $g_i g_j^{-1}$ are in the directions, respectively,

$$\theta_i = \arctan\left(\sin \gamma_{ij}, -\cos \gamma_{ij}\right) = \gamma_{ij} - \frac{\pi}{2},$$

$$\theta_j = \arctan\left(-\sin \alpha_{ij}, -\cos \alpha_{ij}\right) = -\alpha_{ij} - \frac{\pi}{2}.$$

We express the f_{ij}'s in terms of $\alpha_{kq}^{(i)}$, $\alpha_{kq}^{(j)}$, and θ_i and θ_j:

$$f_{ij}(\theta_i, \theta_j) := f_{ij}\left(g_i g_j^{-1}\right)$$

$$= \left\| \sum_{k=-k_{\max}}^{k_{\max}} \sum_{q=1}^{p_k} \left(\alpha_{kq}^{(i)} \psi_{kq}^{(c)}(r, \theta_i) - \alpha_{kq}^{(j)} \psi_{kq}^{(c)}(r, \theta_j) \right) \right\|_{L^2}^2$$

$$= \sum_{k,k',q,q'} c N_{kq} N_{k'q'} \left(\alpha_{kq}^{(i)} e^{ik\theta_i} - \alpha_{kq}^{(j)} e^{ik\theta_j} \right) \left(\alpha_{k'q'}^{(i)} e^{ik'\theta_i} - \alpha_{k'q'}^{(j)} e^{ik'\theta_j} \right)^*$$

$$\cdot \int_0^1 J_k(R_{kq}r) J_{k'}(R_{k'q'}r) \mathrm{d}r.$$

For each k, k', q, q', we approximate the integral

$$\int_0^1 J_k(R_{kq}r) J_{k'}(R_{k'q'}r) \mathrm{d}r$$

with a Gaussian quadrature.

Using the approximation above, we have

$$
f_{ij}(\theta_i, \theta_j) \approx \sum_{k,q,k',q'} b_{k,q,k',q'} \left(\alpha_{kq}^{(i)} \overline{\alpha_{k'q'}^{(i)}} e^{i(k-k')\theta_i} + \alpha_{kq}^{(j)} \overline{\alpha_{k'q'}^{(j)}} e^{i(k-k')\theta_j} \right.
$$
$$
\left. - \alpha_{kq}^{(i)} \overline{\alpha_{k'q'}^{(j)}} e^{i(k\theta_i - k'\theta_j)} - \alpha_{kq}^{(j)} \overline{\alpha_{k'q'}^{(i)}} e^{i(k\theta_j - k'\theta_i)} \right),
$$

where

$$
b_{k,q,k',q'} = c N_{kq} N_{k'q'} \sum_i w_i J_k(R_{kq} r_i) J_{k'}(R_{k'q'} r_i).
$$

In terms of the Euler $(Z-Y-Z)$ angles,

$$
f_{ij}(\alpha_{ij}, \gamma_{ij}) := f_{ij}(\theta_i, \theta_j)
$$
$$
\approx \sum_{k,q,k',q'} b_{k,q,k',q'} e^{-i\frac{\pi}{2}(k-k')} \left(\alpha_{kq}^{(i)} \overline{\alpha_{k'q'}^{(i)}} e^{i(k-k')\gamma_{ij}} + \alpha_{kq}^{(j)} \overline{\alpha_{k'q'}^{(j)}} e^{-i(k-k')\alpha_{ij}} \right.
$$
$$
\left. - \alpha_{kq}^{(i)} \overline{\alpha_{k'q'}^{(j)}} e^{ik\gamma_{ij} + ik'\alpha_{ij}} - \alpha_{kq}^{(j)} \overline{\alpha_{k'q'}^{(i)}} e^{-ik\alpha_{ij} - ik'\gamma_{ij}} \right).
$$

The Fourier coefficients are given by

$$
\hat{f}_{ij}(k) = \int_{SO(3)} f_{ij}(\alpha, \gamma) \left(W^{(k)}(\alpha, \beta, \gamma) \right)^* dg
$$
$$
= \int_0^{2\pi} \int_0^{2\pi} f_{ij}(\alpha, \gamma) \left(\int_0^\pi W^{(k)}(-\gamma + \pi, \beta, -\alpha + \pi) \sin \beta d\beta \right) d\alpha d\gamma.
$$

Note that

$$
\left(\int_0^\pi W^{(k)}(\alpha, \beta, \gamma) \sin \beta d\beta \right)_{m,m'} = \int_0^\pi e^{im\alpha} w_{m,m'}^{(k)}(\beta) e^{im'\gamma} \sin \beta d\beta,
$$

where $w^{(k)}$ is the Wigner *D*-matrix. Let us define

$$
H_k(m, m') := (-1)^{m+m'} \int_0^\pi w_{m,m'}^{(k)}(\beta) \sin \beta d\beta
$$

$$
= (-1)^{m+m'} \int_0^\pi i^{m-m'} \sum_{l=-k}^k w_{l,m}^{(k)}(\pi/2) e^{-il\beta} w_{l,m'}^{(k)}(\pi/2) \sin \beta d\beta
$$

$$
= (-1)^{m+m'} i^{m-m'} \left[2 w_{0,m}^{(k)}(\pi/2) w_{0,m'}^{(k)}(\pi/2) \right.
$$

$$
- \frac{i\pi}{2} w_{1,m}^{(k)}(\pi/2) w_{1,m'}^{(k)}(\pi/2) + \frac{i\pi}{2} w_{-1,m}^{(k)}(\pi/2) w_{-1,m'}^{(k)}(\pi/2)
$$

$$
\left. + \sum_{|l| \geqslant 2}^k \left(\frac{1 + e^{-il\pi}}{1 - l^2} \right) w_{l,m}^{(k)}(\pi/2) w_{l,m'}^{(k)}(\pi/2) \right].
$$

The (m, m')th entry of $\hat{f}_{ij}(k)$ is approximated by

$$
\left(\hat{f}_{ij}(k)\right)_{m,m'} = H_k(m, m') \int_0^{2\pi} \int_0^{2\pi} f_{ij}(\alpha, \gamma) e^{-im\gamma} e^{-im'\alpha} d\alpha d\gamma
$$

$$
= 4\pi^2 H_k(m, m') \sum_{k,q,k',q'} b_{k,q,k',q'} \left(\alpha_{kq}^{(i)} \overline{\alpha_{k'q'}^{(i)}} \delta_{\{m=k-k'\}} \delta_{\{m'=0\}} \right.
$$

$$
+ \alpha_{kq}^{(j)} \overline{\alpha_{k'q'}^{(j)}} \delta_{\{m=0\}} \delta_{\{m'=k'-k\}} - \alpha_{kq}^{(i)} \overline{\alpha_{k'q'}^{(j)}} \delta_{\{m=k\}} \delta_{\{m'=k'\}}
$$

$$
\left. - \alpha_{kq}^{(j)} \overline{\alpha_{k'q'}^{(i)}} \delta_{\{m=-k'\}} \delta_{\{m'=-k\}} \right).
$$

Here, δ is the Kronecker delta and $b_{k,q,k',q'}$ absorbed $e^{-i\frac{\pi}{2}(k-k')}$.

5.3.1. Handedness ambiguity in cryo-EM.

There exists one additional issue specifically for the cryo-EM problem arising from the handedness ambiguity. Suppose an image was projected from some molecular density ϕ and orientation R. Let J be the reflection operator across the imaging plane. The molecular density $J\phi$ under orientation JRJ^* would produce the same projection. In other words, the set of projection images can belong to different molecular densities ϕ and $J\phi$ under different rotations.

We will formalize and deal with the handedness ambiguity in terms of the Wigner D-matrices. Recall that the Wigner D-matrix $W^{(k)}(g)$ corresponding to $(\alpha, \beta, \gamma) \in SO(3)$ is

$$
W^{(k)}(g) = \left[e^{im\alpha} w_{m,m'}^{(k)}(\beta) e^{im\gamma} \right]_{m,m}' = -k^k.
$$

Let $J^{(k)}$ be the following $(2k + 1) \times (2k + 1)$ diagonal matrix:

$$
J^{(k)} := \begin{bmatrix} \ddots & & & & \\ & -1 & & & \\ & & 1 & & \\ & & & -1 & \\ & & & & \ddots \end{bmatrix}.
$$

(The diagonal alternates between $+1$ and -1.) Due to the handedness ambiguity, if $\left\{ W^{(k)}(g_i) W^{(k)}(g_j^{-1}) \right\}_k$ is a solution to (4.4), then $\left\{ J^{(k)} W^{(k)}(g_i) W^{(k)}(g_j^{-1}) J^{(k)} \right\}_k$ is also a valid solution to (4.4). In fact, for any $h \in [0, 1]$,

$$
\left\{ hW^{(k)}(g_i) W^{(k)}(g_j^{-1}) + (1 - h) J^{(k)} W^{(k)}(g_i) W^{(k)}(g_j^{-1}) J^{(k)} \right\}_k
$$

is a valid solution to (4.4).

Let us remove this extra degree of freedom h. Observe that for $h = \frac{1}{2}$,

$$
\frac{1}{2} W^{(k)}(g) + \frac{1}{2} J^{(k)} W^{(k)}(g) J^{(k)} = \begin{cases} e^{im\alpha} w_{m,m'}^{(k)}(\beta) e^{im'\gamma}, & m + m' \equiv 0 \bmod 2, \\ 0, & \text{otherwise.} \end{cases}
$$

I.e., the odd-indexed entries are 0. For example, in the case of $k = 1$,

$$\frac{1}{2}W^{(1)}(g) + \frac{1}{2}J^{(1)}W^{(1)}(g)J^{(1)}$$

$$= \begin{bmatrix} e^{-i\alpha}w^{(1)}_{-1,-1}(\beta)e^{-i\gamma} & 0 & e^{-i\alpha}w^{(1)}_{-1,1}(\beta)e^{i\gamma} \\ 0 & w^{(1)}_{0,0}(\beta) & 0 \\ e^{i\alpha}w^{(1)}_{1,-1}(\beta)e^{-i\gamma} & 0 & e^{i\alpha}w^{(1)}_{1,1}(\beta)e^{i\gamma} \end{bmatrix},$$

and in the case of $k = 2$,

$$\frac{1}{2}W^{(2)}(g) + \frac{1}{2}J^{(2)}W^{(2)}(g)J^{(2)}$$

$$= \begin{bmatrix} e^{-2i\alpha}w^{(2)}_{-2,-2}(\beta)e^{-2i\gamma} & 0 & e^{-2i\alpha}w^{(2)}_{-2,0}(\beta) & 0 & e^{-2i\alpha}w^{(2)}_{-2,2}(\beta)e^{2i\gamma} \\ 0 & e^{-i\alpha}w^{(2)}_{-1,-1}(\beta)e^{-i\gamma} & 0 & e^{-i\alpha}w^{(2)}_{-1,1}(\beta)e^{i\gamma} & 0 \\ w^{(2)}_{0,-2}(\beta)e^{-2i\gamma} & 0 & w^{(2)}_{0,0}(\beta) & 0 & w^{(2)}_{0,2}(\beta)e^{2i\gamma} \\ 0 & e^{i\alpha}w^{(2)}_{1,-1}(\beta)e^{-i\gamma} & 0 & e^{i\alpha}w^{(2)}_{1,1}(\beta)e^{i\gamma} & 0 \\ e^{2i\alpha}w^{(2)}_{2,-2}(\beta)e^{-2i\gamma} & 0 & e^{2i\alpha}w^{(2)}_{2,0}(\beta) & 0 & e^{2i\alpha}w^{(2)}_{2,2}(\beta)e^{2i\gamma} \end{bmatrix}.$$

We constrain the odd-indexed entries of $X^{(k)}_{ij}$'s to be 0 so that the SDP finds the solution with $h = \frac{1}{2}$. Note that, in practice, we do not explicitly add this constraint. Instead, we permute $X^{(k)}_{ij}$ into two disjoint diagonal blocks. For example, in the case of $k = 1$,

$$\frac{1}{2}W^{(1)}(g) + \frac{1}{2}J^{(1)}W^{(1)}(g)J^{(1)}$$

$$= \begin{bmatrix} w^{(1)}_{0,0}(\beta) & 0 & 0 \\ 0 & e^{-i\alpha}w^{(1)}_{-1,-1}(\beta)e^{-i\gamma} & e^{-i\alpha}w^{(1)}_{-1,1}(\beta)e^{i\gamma} \\ 0 & e^{i\alpha}w^{(1)}_{1,-1}(\beta)e^{-i\gamma} & e^{i\alpha}w^{(1)}_{1,1}(\beta)e^{i\gamma} \end{bmatrix},$$

and in the case of $k = 2$,

$$\frac{1}{2}W^{(2)}(g) + \frac{1}{2}J^{(2)}W^{(2)}(g)J^{(2)}$$

$$= \begin{bmatrix} e^{-i\alpha}w^{(2)}_{-1,-1}(\beta)e^{-i\gamma} & e^{-i\alpha}w^{(2)}_{-1,1}(\beta)e^{i\gamma} & 0 & 0 & 0 \\ e^{i\alpha}w^{(2)}_{1,-1}(\beta)e^{-i\gamma} & e^{i\alpha}w^{(2)}_{1,1}(\beta)e^{i\gamma} & 0 & 0 & 0 \\ 0 & 0 & e^{-2i\alpha}w^{(2)}_{-2,-2}(\beta)e^{-2i\gamma} & e^{-2i\alpha}w^{(2)}_{-2,0}(\beta) & e^{-2i\alpha}w^{(2)}_{-2,2}(\beta)e^{2i\gamma} \\ 0 & 0 & w^{(2)}_{0,-2}(\beta)e^{-2i\gamma} & w^{(2)}_{0,0}(\beta) & w^{(2)}_{0,2}(\beta)e^{2i\gamma} \\ 0 & 0 & e^{2i\alpha}w^{(2)}_{2,-2}(\beta)e^{-2i\gamma} & e^{2i\alpha}w^{(2)}_{2,0}(\beta) & e^{2i\alpha}w^{(2)}_{2,2}(\beta)e^{2i\gamma} \end{bmatrix}.$$

We can conjugate each $X^{(k)}_{ij}$ in (4.4) by a permutation and get

$$X^{(k)}_{ij} = \begin{bmatrix} X^{(k,0)}_{ij} & 0 \\ 0 & X^{(k,1)}_{ij} \end{bmatrix}.$$

Similarly, we can conjugate $X^{(k)}$ by a permutation and get

$$X^{(k)} = \begin{bmatrix} X^{(k,0)} & 0 \\ 0 & X^{(k,1)} \end{bmatrix},$$

25

where

$$
X^{(k,0)} = \begin{bmatrix} X_{11}^{(k,0)} & \cdots & X_{1n}^{(k,0)} \\ \vdots & \ddots & \vdots \\ X_{n1}^{(k,0)} & \cdots & X_{nn}^{(k,0)} \end{bmatrix}, \quad X^{(k,1)} = \begin{bmatrix} X_{11}^{(k,1)} & \cdots & X_{1n}^{(k,1)} \\ \vdots & \ddots & \vdots \\ X_{n1}^{(k,1)} & \cdots & X_{nn}^{(k,1)} \end{bmatrix}.
$$

Let us denote the above permutation as Π_k. The objective function in (4.4) is preserved if we conjugate both the $C^{(k)}$'s and the $X^{(k)}$'s by Π_k. I.e.,

$$
\mathrm{tr}\left[C^{(k)}X^{(k)}\right] = \mathrm{tr}\left[\Pi_k C^{(k)}\Pi_k^T \Pi_k X^{(k)}\Pi_k^T\right] = \mathrm{tr}\left[C^{(k,0)}X^{(k,0)}\right] + \mathrm{tr}\left[C^{(k,1)}X^{(k,1)}\right],
$$

where $C^{(k,0)}$ and $C^{(k,1)}$ are blocks corresponding to $X^{(k,0)}$ and $X^{(k,1)}$, respectively. We apply the same permutation to the $X^{(k)}$'s in the constraints of (4.4) and reduce (4.4) to

$$
\begin{aligned}
&\underset{X^{(k,0)},X^{(k,1)}}{\text{minimize}} \quad \sum_{k=0}^{t} \mathrm{tr}\left[C^{(k,0)}X^{(k,0)}\right] + \mathrm{tr}\left[C^{(k,1)}X^{(k,1)}\right] \\
&\text{subject to} \quad X^{(k,0)} \succeq 0, \quad X^{(k,1)} \succeq 0 \\
&\qquad\qquad X_{ii}^{(k,0)} = I_{k\times k}, \quad X_{ii}^{(k,1)} = I_{(k+1)\times(k+1)} \\
&\qquad\qquad \sum_{k=0}^{t} \frac{(3)_{t-k}}{(t-k)!}\left(k+\frac{1}{2}\right)\left(\mathrm{tr}\left[\left(W^{(k,0)}(\alpha,\beta,\gamma)\right)^* X_{ij}^{(k,0)}\right]\right. \qquad\qquad (5.1) \\
&\qquad\qquad \left. +\mathrm{tr}\left[\left(W^{(k,1)}(\alpha,\beta,\gamma)\right)^* X_{ij}^{(k,1)}\right]\right) \geqslant 0 \quad \forall(\alpha,\beta,\gamma) \in \Omega_t \\
&\qquad\qquad X_{ij}^{(0,1)} = 1 \\
&\qquad\qquad Q_{ij} = \mathcal{A}_{E_q}\left(X_{ij}^{(1,0)}, X_{ij}^{(1,1)}\right), \quad Q_{ij} \succeq 0, \quad \mathrm{tr}[Q_{ij}] = 1,
\end{aligned}
$$

where $\mathcal{A}_{E_q}$ defines the linear relationship between Q_{ij} and $\left(X^{(1,0)}, X^{(1,1)}\right)$ as

$$
Y^{(ij)} = \begin{bmatrix} -\dfrac{\mathrm{i}}{\sqrt{2}} & 0 & \dfrac{1}{\sqrt{2}} \\ 0 & 1 & 0 \\ -\dfrac{\mathrm{i}}{\sqrt{2}} & 0 & -\dfrac{1}{\sqrt{2}} \end{bmatrix}^* \begin{bmatrix} X_{ij}^{(1,1)}(1,1) & 0 & X_{ij}^{(1,1)}(1,2) \\ 0 & X_{ij}^{(1,0)} & 0 \\ X_{ij}^{(1,1)}(2,1) & 0 & X_{ij}^{(1,1)}(2,2) \end{bmatrix} \begin{bmatrix} -\dfrac{\mathrm{i}}{\sqrt{2}} & 0 & \dfrac{1}{\sqrt{2}} \\ 0 & 1 & 0 \\ -\dfrac{\mathrm{i}}{\sqrt{2}} & 0 & -\dfrac{1}{\sqrt{2}} \end{bmatrix},
$$

$$
Q_{ij} = \frac{1}{4}\begin{bmatrix} 1+Y_{11}^{(ij)}+Y_{22}^{(ij)}+Y_{33}^{(ij)} & 0 & 0 & -(Y_{31}^{(ij)}-Y_{13}^{(ij)}) \\ 0 & 1-Y_{11}^{(ij)}-Y_{22}^{(ij)}+Y_{33}^{(ij)} & -(Y_{31}^{(ij)}+Y_{13}^{(ij)}) & 0 \\ 0 & -(Y_{31}^{(ij)}+Y_{13}^{(ij)}) & 1+Y_{11}^{(ij)}-Y_{22}^{(ij)}-Y_{33}^{(ij)} & 0 \\ -(Y_{31}^{(ij)}-Y_{13}^{(ij)}) & 0 & 0 & 1-Y_{11}^{(ij)}+Y_{22}^{(ij)}-Y_{33}^{(ij)} \end{bmatrix}.
$$

5.3.2. Rounding. (5.1) gives us the $X^{(k,0)}$'s and the $X^{(k,1)}$'s. We aggregate $X^{(1,0)}$ and $X^{(1,1)}$ into the $3n \times 3n$ matrix $X^{(1)}$. From the $X^{(1)}$'s, we build the *synchronization matrix S* described in [18] and apply the `cryo_syncrotations` function on S from the ASPIRE software package [42] to recover $(\alpha_1, \beta_1, \gamma_1), \ldots, (\alpha_n, \beta_n, \gamma_n)$. Note that we do not truncate the eigenvectors and eigenvalues from $X^{(1)}$.
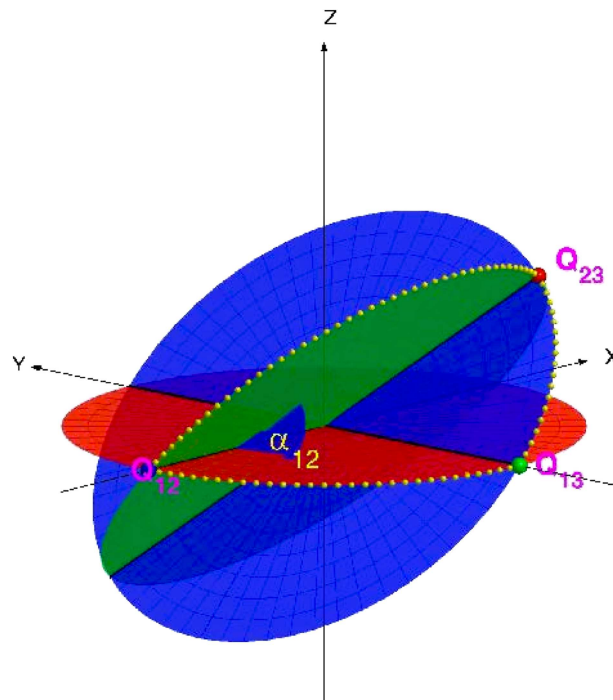
Figure 6. Three cryo-EM images uniquely determine their orientations up to handedness. Image courtesy of Singer *et al* [46].

6. Implementation and results for synthetic cryo-EM datasets

In this section, we give a brief history of methods used for determining orientations of cryo-EM projection images, and where we stand against the current state-of-the-art.

In 1986, Vainshtein and Goncharov developed a common-lines based method for *ab initio* modeling in cryo-EM [29]. In 1987, Van Heel also independently discovered the same method, and coined it *angular reconstitution* [28]. Recall that by the Fourier slice theorem, two cryo-EM projection images (in Fourier space) must intersect along a common line. Given three cryo-EM images from different viewing directions, their common lines must uniquely determine their relative orientations up to handedness. (See figure 6.) The orientations of the rest of the images are determined by common lines with the first three images. This is angular reconstitution in a nutshell.

In 1992, Farrow and Ottensmeyer expanded upon angular reconstitution by developing a method to sequentially add images via least squares [43]. One major drawback in sequentially assigning orientations of cryo-EM images is the propagation of error due to false common line detection. In 1996, Penczek, Zhu, and Frank tried to circumvent the issue via brute-force search for a global energy minimizer [44]. However, the search space is simply too big for that method to be applicable. In 2006, Mallick *et al* introduced a Bayesian method in which common lines between pairs of images are determined by their common lines with different projection triplets [45]. In the method by Mallick *et al*, at least seven common lines must be correctly and simultaneously determined, which can be problematic. In 2010, Singer *et al* lowered the requirement to that only two common lines need to be correctly and simultaneously
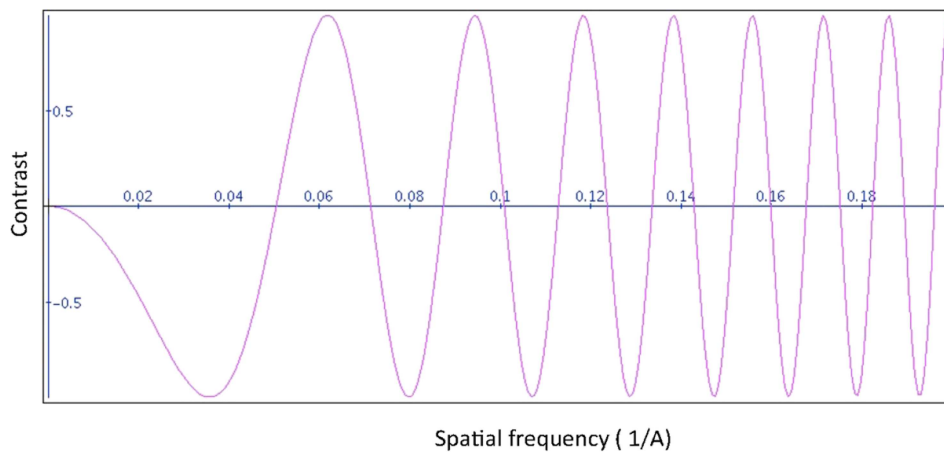
Figure 7. Example created by Jiang and Chiu available at http://jiang.bio.purdue.edu/software/ctf/ctfapplet.html.

determined [46]. In 2011, Singer and Shkolnisky built upon [46] by adding a global consistency condition among the orientations [18]. This method is called *synchronization*, and it is regarded as the current state-of-the-art.

Note that all avaible methods for orientation determination, including the NUG approach proposed in this paper, cannot be directly applied to raw experimental images. We will explain the factors preventing us from doing so in the next subsection. We therefore numerically validate and evaluate the method using synthetic data.

6.1. Shifts, CTF and contrast

In comparison to the simplistic forward model (2.8), there are three major imperfections in the experimental cryo-EM datasets. First, the images are not centered, and the common lines will not correspond exactly even under their true orientations. Second, the images are subject to the contrast transfer function (CTF) of the electron microscope. A CTF, as a function of radial frequency, is shown in figure 7. Making matters mathematically more challenging, a CTF is typically estimated per micrograph and each micrograph would have a different CTF. Thus, all images from the same micrograph are typically assigned the same CTF. We also say those images belong to the same *defocus* group. However, as shown in table 2 of section 6.4, shifts and CTFs are not detrimental to NUG's performance. This brings us to the third obstacle, which does prevent us from directly applying NUG on raw experimental images. The ice layer in which the molecules are frozen is not uniform. It can be thicker/thinner where different projections are taken, and this effect is equivalent to scaling the projections by a factor $\gamma > 0$. So, various projections have various contrasts. Those effects are typically mitigated in class averages. Class averages are formed by in-plane aligning and averaging raw images that are estimated to have similar viewing directions. It is possible to apply NUG on class averages instead of the original raw experimental images. However, the quality of the results then depends crucially on the specific class averaging procedure being used and does not provide much insight into the performance of NUG itself.

6.2. ADMM implementation

There are two parts that are particularly challenging for obtaining a numerical solution to the NUG SDP (4.4) and (5.1):

- implementing an SDP solver that is scalable to real-world problems such as orientation estimation in cryo-EM,
- computing the coefficient matrix for generic objective function f_{ij}'s.

For the cryo-EM problem, the NUG SDP is simply too big to iterate on solvers based on techniques like interior point methods. Interior point methods are known for their accuracy. However, their accuracy is achieved at the expense of computational complexity. In essence, interior point methods solve a system of linear equations that attempts to satisfy both primal and dual feasibilities [47]. That is, solvers based on interior point methods have to invert a matrix that contains both primal and dual variables. The number and size of the signals coupled with the number of inequality constraints in (4.4) and (5.1) make this inversion impractical. Instead, we use the alternating direction method of multipliers, or ADMM. As the name suggests, ADMM alternates between the primal and the different sets of dual variables. More importantly, the steps in ADMM are linear, with the exception of an eigendecomposition on the primal variable. In practice, the eigendecomposition on the primal variable is manageable.

In general, it is not possible to obtain a closed-form expression for the coefficient matrices $C^{(k)}$'s. Sometimes, we need to employ numerical integration schemes over the group $\mathcal{G}$ to find the $C^{(k)}$'s.

6.2.1. The alternating direction method of multipliers.
In short, ADMM solves the augmented Lagrangian via iterative partial updating. We solve an unconstrained optimization problem over the objective variable, and enforce the constraints via dual variables. We will express the NUG SDP in a more general form, and derive ADMM updates in the more general setting. (4.4) and (5.1) are SDPs of the following form:

$$
\begin{aligned}
\underset{X}{\text{minimize}} \quad & \langle C, X \rangle \\
\text{subject to} \quad & X \succeq 0 \\
& \mathcal{A}_E(X) = b_E \\
& \mathcal{A}_I(X) \geqslant b_I.
\end{aligned}
\tag{6.1}
$$

We can think of C and X as the following block matrices:

$$
C = \begin{bmatrix} C^{(0)} & & & \\ & C^{(1)} & & \\ & & \ddots & \\ & & & C^{(t)} \end{bmatrix}, \quad X = \begin{bmatrix} X^{(0)} & & & \\ & X^{(1)} & & \\ & & \ddots & \\ & & & X^{(t)} \end{bmatrix}.
$$

Note that going from (4.4) and (5.1) to (6.1), we used the fact that

$$
\text{tr}\left[C^{(k)} X^{(k)} \right] = \left\langle C^{(k)}, X^{(k)} \right\rangle
$$

because the $C^{(k)}$'s are Hermitian. $\mathcal{A}_E$ and $\mathcal{A}_I$ are linear operators that encapsulate the equality and inequality constraints, respectively. For example, we can think of $\mathcal{A}$ as

$$
\mathcal{A}(X) = \begin{bmatrix} \langle A_I^{(1)}, X \rangle \\ \vdots \\ \langle A_I^{(m)}, X \rangle \\ \vdots \end{bmatrix},
$$

where, for the constraint $\sum_{k=0}^{t} b_k \mathrm{tr}\left[\rho_k^*(g_m) X_{ij}^{(k)} \right]$, $A_I^{(m)}$ is the matrix containing $b_k \rho_k(g_m)$ at the position of $X_{ij}^{(k)}$ and 0 everywhere else.

More concretely, $\mathcal{A}(X) = b$ is equivalent to $A \, \mathrm{vec}(X) = b$, where

$$
A = \left[\mathrm{vec}(A^{(1)}) \cdots \mathrm{vec}(A^{(m)}) \cdots \right]^T
$$

and vec vectorizes the matrix $A^{(m)}$ along the columns. The adjoint operator of $\mathcal{A}$ is given by

$$
\mathcal{A}^*(z) = \mathrm{mat}(A^T z),
$$

where mat is the reverse operator of vec. Furthermore, we can verify

- $\langle \mathcal{A}(X), z \rangle = \langle A \, \mathrm{vec}(X), z \rangle = \langle \mathrm{mat}(A^T z), X \rangle = \langle \mathcal{A}^*(z), X \rangle$,
- $(\mathcal{A}\mathcal{A}^*)(z) = \mathcal{A}(\mathcal{A}^*(z)) = (AA^T)z$,
- $(\mathcal{A}^*\mathcal{A})(z) = \mathcal{A}^*(\mathcal{A}(X)) = \mathrm{mat}\left((A^T A)\mathrm{vec}(X) \right)$.

Now, we describe the ADMM solver outlined in [48] for SDP (6.1). ADMM is essentially a series of partial iterative updates based on the augmented Lagrangian. So, we will define the dual variables and write out the augmented Lagrangian. Then, we derive the updates by setting the gradient of the augmented Lagrangian, with respect to specific variables, to 0.

The dual variables corresponding to (6.1) are

$$
\begin{aligned}
-X \preceq 0 &\qquad \leftrightarrow \qquad S \succeq 0 \\
b_E - \mathcal{A}_E(X) = 0 &\qquad \leftrightarrow \qquad y_E \\
b_I - \mathcal{A}_I(X) \leqslant 0 &\qquad \leftrightarrow \qquad y_I \geqslant 0.
\end{aligned}
$$

The Lagrangian for (6.1) is

$$
L(X, S, y_E, y_I) := -\left\langle (b_E, b_I), (y_E, y_I) \right\rangle + \left\langle X, S + \mathcal{A}_E^*(y_E) + \mathcal{A}_I^*(y_I) - C \right\rangle. \quad (6.2)
$$

The motivation for defining (6.2) is so that the primal variable X satisfies the constraints using the dual variables via

$$
\underset{S \succeq 0, y_E, y_I \geqslant b_I}{\mathrm{minimize}} \quad \underset{X}{\mathrm{maximize}} \quad L(X, S, y_E, y_I). \quad (6.3)
$$

Notice that (6.3) is an unconstrained optimization problem over X. If X is not PSD, then there exists $S \succeq 0$ such that $L(X, S, y_E, y_I) = -\infty$. Thus, due to the inner maximization over X, (6.3) produces a solution satisfying $X \succeq 0$. The equality and inequality constraints are enforced in a similar manner. The maximizing X must satisfy

$$
\nabla_X L = 0 \Leftrightarrow S + \mathcal{A}_E^*(y_E) + \mathcal{A}_I^*(y_I) = C.
$$

In fact, this gives us the dual problem to (6.1)

$$\underset{S, y_E, y_I}{\text{minimize}} \quad \delta_{\preceq 0}(S) + \delta_{\leqslant 0}(y_I) - \langle (b_E, b_I), (y_E, y_I) \rangle$$

$$\text{subject to} \quad S + \mathcal{A}_E^*(y_E) + \mathcal{A}_I^*(y_I) = C.$$

The augmented Lagrangian is the Lagrangian with the dual variables regularized by the Frobenius norm:

$$L_\rho(X, S, y_E, y_I) := - \langle (b_E, b_I), (y_E, y_I) \rangle + \langle X, S + \mathcal{A}_E^*(y_E) + \mathcal{A}_I^*(y_I) - C \rangle$$

$$+ \frac{\rho}{2} \| S + \mathcal{A}_E^*(y_E) + \mathcal{A}_I^*(y_I) - C \|_{\text{Fro}}^2 . \tag{6.4}$$

This regularization term is crucial for numerical convergence of the solver.

The updates for the dual variables are given by their individual optimality conditions.

- Solving $0 = \nabla_S L_\rho$ for S and applying the PSD projection, we get

$$S^{(k+1)} = \left[C - \mathcal{A}_E^*(y_E^{(k)}) - \mathcal{A}_I^*(y_I^{(k)}) - \frac{1}{\rho} X^{(k)} \right]_{\succeq 0}.$$

- Solving $0 = \nabla_{y_E} L_\rho$ for y_E, we get

$$y_E^{(k+1)} = \left(\mathcal{A}_E \mathcal{A}_E^* \right)^{-1} \left(\frac{1}{\rho} \left(b_E - \mathcal{A}_E(X^{(k)}) \right) - \mathcal{A}_E \left(\mathcal{A}_I^*(y_I^{(k)}) + S^{(k)} - C \right) \right).$$

Note that for the NUG SDP, $\mathcal{A}_E \mathcal{A}_E^* = \mathcal{I}$ because each equality constraint is on a single entry in X. So, this update simplifies to

$$y_E^{(k+1)} = \frac{1}{\rho} \left(b_E - \mathcal{A}_E(X^{(k)}) \right) - \mathcal{A}_E \left(\mathcal{A}_I^*(y_I^{(k)}) + S^{(k)} - C \right).$$

- Note that if we solve $0 = \nabla_{y_I} L_\rho$ for y_I, we will have to invert the operator $\mathcal{A}_I \mathcal{A}_I^*$, and this is extremely computationally challenging. Instead, we add an additional regularization term $\frac{\rho}{2} \| y_I - y_I^{(k)} \|_{\lambda \mathcal{I} - \mathcal{A}_I \mathcal{A}_I^*}^2$, where λ is the largest eigenvalue of $\mathcal{A}_I \mathcal{A}_I^*$ and $y_I^{(k)}$ is the element from the previous y_I-update. Solving $0 = \nabla_{y_I} \left[L_\rho + \frac{\rho}{2} \| y_I - y_I^{(k)} \|_{\lambda \mathcal{I} - \mathcal{A}_I \mathcal{A}_I^*}^2 \right]$ for y_I and applying the the nonnegativity projection, we get

$$y_I^{(k+1)} = \left[\frac{1}{\rho \lambda} \left(b_I - \mathcal{A}_I(X^{(k)}) \right) \right.$$

$$\left. + \frac{1}{\lambda} \mathcal{A}_I \left(C - \mathcal{A}_E^*(y_E^{(k)}) - \mathcal{A}_I^*(y_I^{(k)}) - S^{(k)} \right) + y_I^{(k)} \right]_{\geqslant 0}.$$

The update for the primal variable is simply a gradient descent given by

$$X^{(k+1)} = X^{(k)} + \rho \left(S^{(k)} + \mathcal{A}_E^*(y_E^{(k)}) + \mathcal{A}_I^*(y_I^{(k)}) - C \right).$$

We initialize the variables as the following:

- $X^{(0)} = I$,
- $y_I^{(0)} = 0$,
- $S^{(0)} = 0$,
- $y_E^{(0)} = \frac{1}{2}\mathcal{A}_E\left(C - \mathcal{A}_I^*(y_I^{(0)}) - S^{(0)}\right)$.

We apply the updates in the following order:

(a) $S^{(k+1)} = \left[C - \mathcal{A}_E^*(y_E^{(k)}) - \mathcal{A}_I^*(y_I^{(k)}) - \frac{1}{\rho}X^{(k)}\right]_{\succeq 0}$,

(b) $y_E^{(k+1/2)} = \mathcal{A}_E\left(C - \mathcal{A}_I^*(y_I^{(k)}) - S^{(k+1)}\right)$,

(c) $y_I^{(k+1)} = \left[\frac{1}{\rho\lambda}\left(b_I - \mathcal{A}_I(X^{(k)})\right) + \frac{1}{\lambda}\mathcal{A}_I\left(C - \mathcal{A}_E^*(y_E^{(k+1/2)}) - \mathcal{A}_I^*(y_I^{(k)}) - S^{(k+1)}\right) + y_I^{(k)}\right]_{\geqslant 0}$,

(d) $y_E^{(k+1)} = \mathcal{A}_E\left(C - \mathcal{A}_I^*(y_I^{(k+1)}) - S^{(k+1)}\right)$,

(e) $X^{(k+1)} = X^{(k)} + \rho\left(S^{(k+1)} + \mathcal{A}_E^*(y_E^{(k+1)}) + \mathcal{A}_I^*(y_I^{(k+1)}) - C\right)$.

Note that we updated y_E twice, which guarantees the solver's convergence to the optimizer [48].

6.2.2. Fourier coefficients for bandlimited functions on SO(3). The coefficient matrix in (3.12) is composed of

$$\hat{f}_{ij}(k) = \int_{\mathcal{G}} f_{ij}(g)\rho_k^*(g)\mathrm{d}g.$$

We obtained closed-form expressions for the registration problem (2.4) under the assumption that the noise in the underlying signal is Gaussian. For the cryo-EM problem (2.11), we require a single approximation of an integral using Gaussian quadrature. However, this is not always the case. For example, the noise in the cryo-EM projections is better modeled by the Poisson distribution [49]. We describe the numerical integration scheme in [50] that can be used to obtain the desired Fourier coefficients $\hat{f}_{ij}(k)$. We need to make the assumption that f_{ij}'s are bandlimited by T. The method described in [50] has a better computational complexity than straight forward numerical integration. We will define the quadrature, and then outline the evaluation over the quadrature.

For function f with bandlimit T, we have the equality

$$\hat{f}(k) = \frac{1}{(2T)^2}\sum_{j_1=0}^{2T-1}\sum_{j_2=0}^{2T-1}\sum_{l=0}^{2T-1} b_T(l) f(\alpha_{j_1}, \beta_l, \gamma_{j_2})\rho_k^*(\alpha_{j_1}, \beta_l, \gamma_{j_2}), \tag{6.5}$$

where

$$b_T(l) = \frac{2}{T}\sin\left[\frac{\pi(2l+1)}{4T}\right]\sum_{m=0}^{T-1}\frac{1}{2m+1}\sin\left[\frac{\pi(2l+1)(2m+1)}{4T}\right],$$

and

$$\alpha_{j_1} = \frac{2\pi j_1}{2T}, \quad \beta_l = \frac{\pi(2l+1)}{4T}, \quad \gamma_{j_2} = \frac{2\pi j_2}{2T}, \quad 0 \leqslant j_1, j_2, l < 2T.$$

Recall that

$$[\rho_k(\alpha, \beta, \gamma)]_{m,m'} = e^{im\alpha} w_{m,m'}^{(k)}(\beta) e^{im'\gamma}.$$

We can re-write the entries in (6.5) as

$$\left[\hat{f}(k)\right]_{m,m'} = \frac{1}{(2T)^2} \sum_{l=0}^{2T-1} b_T(l) w_{m',m}^{(k)}(\beta_l) \sum_{j_2=0}^{2T-1} e^{-im\gamma_{j_2}} \sum_{j_1=0}^{2T-1} e^{-im'\alpha_{j_1}} f(\alpha_{j_1}, \beta_l, \gamma_{j_2}).$$

(6.6)

By rearranging the terms in (6.6), it becomes obvious that we should compute $\hat{f}(k)$ in the following order:

(a) for all $0 \leqslant j_2, l < 2T$ and $-T \leqslant m' \leqslant T$, compute

$$S_1(l, j_2, m') = \frac{1}{2T} \sum_{j_1=0}^{2T-1} e^{-im'\alpha_{j_1}} f(\alpha_{j_1}, \beta_l, \gamma_{j_2}),$$

(b) for all $0 \leqslant l < 2T$ and $-T \leqslant m', m \leqslant T$, compute

$$S_2(l, m', m) = \frac{1}{2T} \sum_{j_2=0}^{2T-1} e^{-im\gamma_{j_2}} S_1(l, j_2, m'),$$

(c) for all $-T \leqslant m', m \leqslant T$, compute

$$\left[\hat{f}(k)\right]_{m,m'} = \sum_{l=0}^{2T-1} b_T(l) w_{m',m}^{(k)}(\beta_l) S_2(l, m', m).$$

The complexity to compute $\left[\hat{f}(k)\right]_{m,m'}$ for all m, m' and k is $\mathcal{O}(T^4)$; and the complexity of the straight-forward evaluation in (6.5) is $\mathcal{O}(T^6)$ [50].

6.2.3. Simplification for cryo-EM. Recall that the objective function for cryo-EM does not depend on β. I.e.,

$$f(\alpha, \gamma) = f(\alpha, \beta_1, \gamma) = f(\alpha, \beta_2, \gamma).$$

(6.6) reduces to

$$\left[\hat{f}(k)\right]_{m,m'} = \left(\sum_{l=0}^{2T-1} b_T(l) w_{m',m}^{(k)}(\beta_l) \right)$$

$$\times \left(\frac{1}{(2T)^2} \sum_{j_2=0}^{2T-1} e^{-im\gamma_{j_2}} \sum_{j_1=0}^{2T-1} e^{-im'\alpha_{j_1}} f(\alpha_{j_1}, \gamma_{j_2}) \right). \quad (6.7)$$

We compute (6.7) in the following order:

(a) compute

$$B_T(k) = \sum_{l=0}^{2T-1} b_T(l) w_{m',m}^{(k)}(\beta_l),$$

33

(b) for all $0 \leqslant j_2 < 2T$ and $-T \leqslant m' \leqslant T$, compute

$$S_1(j_2, m') = \frac{1}{2T} \sum_{j_1=0}^{2T-1} e^{-im'\alpha_{j_1}} f(\alpha_{j_1}, \gamma_{j_2}),$$

(c) for all $-T \leqslant m', m \leqslant T$, compute

$$S_2(m', m) = \frac{1}{2T} \sum_{j_2=0}^{2T-1} e^{-im\gamma_{j_2}} S_1(j_2, m'),$$

(d) for all $-T \leqslant m', m \leqslant T$, compute

$$\left[\hat{f}(k)\right]_{m,m'} = B_T(k)S_2(m', m).$$

The complexity to compute $\left[\hat{f}(k)\right]_{m,m'}$ for all m, m' and k is $\mathcal{O}(T^3)$.

6.3. Rotation MSE and FSC resolution

We assess the performance of orientation estimation method using the mean squared error (MSE), defined as follows:

$$\text{MSE} := \frac{1}{n} \sum_{i=1}^{n} \|\hat{R}_i - R_i\|_{\text{Fro}}^2. \tag{6.8}$$

Here R_i are the true rotations (which are known in the simulation setting) and $\hat{R}_i$ are the estimated rotations (note that we previously used the hat notation for the Fourier transform, but here it is used for estimators). Since the estimation is up to a global rotation and possibly handedness, the two sets of rotations are aligned prior to computing the MSE.

In addition, we will include the *Fourier shell correlation* (FSC) of the reconstructed structure against the known structure. We reconstruct the three-dimensional structure (in Fourier space) using the estimated orientations to get $\hat{\phi}$, and compare it against the known structure ϕ (in Fourier space). For each spatial frequency r_k, we calculate the FSC

$$\text{FSC}(r_k) := \frac{\displaystyle\sum_{\|v\|_2 = r_k} \hat{\phi}(v)\phi^*(v)}{\displaystyle\sum_{\|v\|_2 = r_k} |\hat{\phi}(v)|^2 \sum_{\|v\|_2 = r_k} |\phi(v)|^2}.$$

We derive the resolution (in Angstroms) by interpolating the FSC until we reach an r_k yielding $\text{FSC}(r_k) \leqslant 0.5$. See [51] for a discussion on the FSC and the cutoff value.

Note that the rotation MSE is the most direct assessment of orientation estimation methods. After all, the stated objective of orientation estimation in cryo-EM is to estimate the orientations. In addition, the quality of the three-dimensional reconstruction depends on several other factors such as the signal-to-noise ratio of the images, the number of images, the distribution of the viewing directions and the CTF of the images. We focus on the rotation MSE because the other factors mentioned are independent of the algorithm being used.

6.4. Simulated data

With the ASPIRE software package in [42], we generate a set of 100 and a set of 500 projection images of size 129×129 from the *Plasmodium falciparum 80S ribosome* (see

Table 1. The resolutions shown are in Angstroms. At high SNR, synchronization is accurate to more decimal places. This is not surprising since we have compromised on various truncations and discretizations for computational speed, etc. However, as the noise increases, we see NUG outperforming synchronization.

SNR	NUG MSE	sync. MSE	NUG res	sync res
		100 images		
1/1	0.0153	1.2759×10^{-4}	24.6	20.8
1/2	0.0155	1.3593×10^{-4}	21.5	20.3
1/4	0.0174	3.6615×10^{-4}	27.0	22.4
1/8	0.0192	0.0037	28.7	25.8
1/16	0.0227	0.0300	29.8	30.9
1/32	0.0298	0.1572	35.6	45.8
1/64	0.1559	2.7818	45.3	174.1
1/128	2.1239	4.1492	97.9	175.3
		500 images		
1/1	0.0125	4.1412×10^{-5}	18.1	14.6
1/2	0.0130	5.1825×10^{-5}	20.8	18.2
1/4	0.0134	1.5268×10^{-4}	21.7	17.0
1/8	0.0143	0.0018	16.4	18.1
1/16	0.0175	0.0189	20.8	17.7
1/32	0.0195	0.1559	24.6	30.7
1/64	0.0460	2.2496	29.3	71.1
1/128	1.6060	3.1661	64.2	107.6

https://ebi.ac.uk/pdbe/emdb/empiar/entry/10028/.) We add Gaussian noise to the simulated projections, and then apply a circular mask to zero out the noise outside of the radius. Table 1 shows a comparison of NUG and the synchronization method for different noise levels. In particular, the estimation error of NUG at SNR = 1/64 with 500 images is sufficiently low (MSE = 0.05) to result in a meaningful three-D *ab initio* model (with <30 Å resolution). Numerical experiments were conducted on a cluster of Intel(R) Xeon(R) CPU E7-8880v3 @ 2.30 GHz totaling 144 cores and 792 GB of RAM. The synchronization method is roughly two orders of magnitude faster than NUG, but NUG gives more accurate estimates at low SNR.

To put the numbers in table 1 into perspective, an MSE of 0.1, for example, can be regarded as small enough in one instance to lead to a good reconstruction, but too large in another. The three-dimensional reconstruction from 100 images at low SNR (e.g., SNR = 1/128) looks bad even if one uses the true orientations (i.e., MSE = 0). On the other hand, the three-dimensional reconstruction from 500 images at moderate SNR (e.g., SNR = 1/32) would look very decent at MSE = 0.1. The latter case is shown in figure 7 in [18], where SNR = 1/32 and the MSE is slightly above 0.1. The resolution measure quantifies the quality of the three-D *ab initio* model. An idea about the quality of three-dimensional reconstruction with respect to the MSE can also be obtained from table 3 and figure 8 in [27]. We would like to point out that for the same computational cost, synchronization can process more images than NUG, and therefore may yield better three-dimensional reconstructions. However, beyond the theoretical importance, NUG still might be useful in situations in which the number of images is limited (e.g., when there are only a handful number of class averages or in electron tomography).

Table 2. Effects of shifts, CTFs and contrast on the MSE (defined in (6.8)) of the NUG SDP cryo-EM. Shifts are sampled from $\mathcal{U}(-5, 5)$, CTFs are drawn uniformly from 4 defocus groups and contrasts are sampled from $\mathcal{U}(0.5, 1.5)$. We used 500 simulated projections of size 64×64.

SNR	benchmark	CTF	CTF and shifts	CTF and contrast
1/4	0.0285	0.0381	0.3382	0.7342
1/8	0.0333	0.1450	0.5741	>2.0
1/16	0.0698	0.7631	1.3391	>2.0
1/32	0.4387	1.9199	>2.0	>2.0
1/64	1.8587	>2.0	>2.0	>2.0
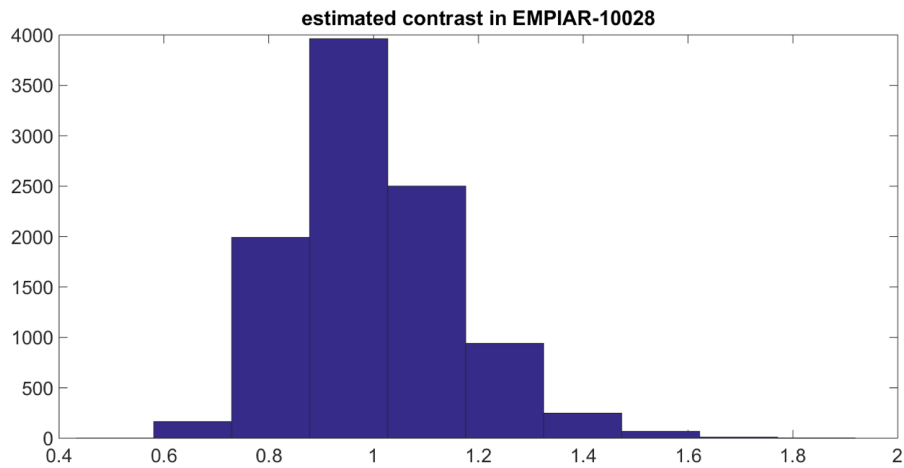
Figure 8. Distribution of contrast in the experimental dataset EMPIAR-10028.

We also illustrate the effects of contrast, CTF and shifts on the performance of NUG in the numerical experiment reported in table 2. In this numerical experiment, phase flipping was applied to correct for the phases of the CTF but not their magnitudes. As for shifts, we simply ignore it in the NUG SDP. In the future, we can expand (4.4) to include shifts to improve upon our estimates. Please see section 6.1 for a discussion on the effects of contrast, CTF and shifts.

The distribution for the contrast was based on the following numerical experiment. We used the publicly available experimental dataset *EMPIAR-10028* found here: https://ebi.ac.uk/pdbe/emdb/empiar/entry/10028/. Using the known three-dimensional structure of the molecule, we construct 10 000 simulated clean projections of size 64×64 at different orientations sampled uniformly over $SO(3)$. For each image $\hat{I}$ in the experimental dataset, we solve for γ in

$$\underset{\gamma, I_i}{\text{minimize}} \quad \|\gamma\hat{I} - I_i\|_{\text{Fro}}^2,$$

where I_i's are simulated projections generated from the known structure. Figure 8 shows the empirical distribution of γ.

7. Summary

The NUG problem consists in the minimization of the sum of pairwise cost functions defined over the ratio between group elements, for arbitrary compact groups. This corresponds to the simultaneous multi-alignment of many datapoints (e.g. signals, images, or molecule densities) with respect to transformations given by an action of the corresponding group. We presented a methodology to solve this problem involving a relaxation of the problem to an SDP and an implementation of an ADMM algorithm to solve the resulting SDP.

In this paper we focused mainly in the context of alignment over $SO(2)$ and $SO(3)$. A notable example is that of finding a consistent set of pairwise rotations of many different functions defined on a sphere which globally minimize the disagreement between pairs of functions.

The NUG problem arises in cryo-EM, where the task is to recover the relative orientations of many noisy two-dimensional projections images of a three-dimensional object, each obtained from a different unknown viewing direction. Once good alignments are estimated, the three-dimensional object can be reconstructed using standard tomography algorithms. In this paper we formulate the problem of image alignment in cryo-EM as an instance of NUG, and demonstrate the applicability to simulated datasets.

The computational and numerical considerations require truncations of the both the cost functions and group representations; a general approach is proposed as part of the formulation as an SDP, and additional methods particularly developed for the case of $SO(3)$, and to the special properties of the cryo-EM problem, are presented.

It is noteworthy that, compared to previous work on related alignment problems, the formulation of the problem as an SDP can provide a certificate for optimality in some cases. Specifically, whenever the solution of the SDP also satisfies the nonconvex constraints that have been relaxed, it serves as a certificate that optimality has been achieved. In numerical work not reported in this paper we have numerically observed optimality of NUG SDP for some instances of MRA, but not for estimation of orientations in cryo-EM. Better theoretical understanding of when NUG SDP achieves optimality is an interesting open problem. In the context of cryo-EM, like other common-line based approaches, it does not require an initial guess. These properties make NUG a candidate for future work on robust *ab initio* alignment which would provide an initialization for other refinement algorithms.

Acknowledgments

ORCID iDs

Yutong Chen ⬤ https://orcid.org/0000-0003-1186-3947
Roy R Lederman ⬤ https://orcid.org/0000-0002-5080-4483

References

[1] Singer A 2011 Angular synchronization by eigenvectors and semidefinite programming *Appl. Comput. Harmon. Anal.* **30** 20–36

[2] Goemans M X and Williamson D P 1995 Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming *Journal of the Association for Computing and Machinery* **42** 1115–45

[3] Charikar M, Makarychev K and Makarychev Y 2006 Near-optimal algorithms for unique games *Proc. of the 38th ACM Symp. on Theory of Computing*

[4] Khot S 2002 On the power of unique 2-prover 1-round games *Thiry-fourth annual ACM symposium on Theory of computing*

[5] Khot S 2010 On the unique games conjecture (invited survey) *Proc. of the 2010 IEEE 25th Annual Conf. on Computational Complexity, CCC '10* (Washington, DC, USA: IEEE Computer Society) pp 99–121

[6] Bandeira A S, Charikar M, Singer A and Zhu A 2014 Multireference alignment using semidefinite programming *5th Innovations in Theoretical Computer Science* (*ITCS 2014*)

[7] Alon N and Naor A 2004 Approximating the cut-norm via Grothendieck's inequality *Proc. of the 36 th ACM STOC* (ACM Press) pp 72–80

[8] Nesterov Y 1998 Semidefinite relaxation and nonconvex quadratic optimization *Optim. Methods Software* **9** 141–60

[9] Abbe E, Bandeira A S and Hall G 2016 Exact recovery in the stochastic block model *IEEE Trans. Inf. Theory* **62** 471–87

[10] Bandeira A S 2018 Random Laplacian matrices and convex relaxations *Found. Comput. Math.* **18** 345–79

[11] Abbe E, Bandeira A S, Bracher A and Singer A 2014 Decoding binary node labels from censored edge measurements: Phase transition and efficient recovery *IEEE Transactions on Network Science and Engineering* **1** 10–22

[12] Cucuringu M 2015 Synchronization over Z2 and community detection in signed multiplex networks with constraints *Journal of Complex Networks* **3** 469–506

[13] Bandeira A S, Kennedy C and Singer A 2016 Approximating the little Grothendieck problem over the orthogonal and unitary groups *Math. Program.* **160** 433–75

[14] Naor A, Regev O and Vidick T 2013 Efficient rounding for the noncommutative Grothendieck inequality *Proc. of the 45th Annual ACM Symp. on Symp. on Theory of Computing, STOC '13* (New York: ACM) pp 71–80

[15] Chen Y, Huang Q-X and Guibas L 2014 Near-optimal joint object matching via convex relaxation *Proceedings of the 31st Int. Conf. on Machine Learning* vol 32 pp 100–8

[16] Huang Q-X and Guibas L 2013 Consistent shape maps via semidefinite programming *Comput. Graph. Forum* **32** 177–86

[17] Lederman R R and Singer A 2019 A representation theory perspective on simultaneous alignment and classification *Appl. Comput. Harmon. Anal.* http://doi.org/10.1016/j.acha.2019.05.005

[18] Shkolnisky Y and Singer A 2012 Viewing direction estimation in cryo-EM using synchronization *SIAM J. Imag. Sci.* **5** 1088–110

[19] Kam Z 1980 The reconstruction of structure from electron micrographs of randomly oriented particles *J. Theor. Biol.* **82** 15–39

[20] Bai X C, McMullan G and Scheres S H 2015 How cryo-EM is revolutionizing structural biology *Trends Biochem. Sci.* **40** 49–57

[21] Wang L and Sigworth F J 2006 Cryo-EM and single particles *Physiology* **21** 13–8

[22] Bandeira A S, Singer A and Spielman D A 2013 A Cheeger inequality for the graph connection Laplacian *SIAM J. Matrix Anal. Appl.* **34** 1611–30

[23] Singer A, Zhao Z, Shkolnisky Y and Hadani R 2011 Viewing angle classification of cryo-Electron Microscopy images using eigenvectors *SIAM J. Imag. Sci.* **4** 723–59

[24] Bannai E and Bannai E 2009 A survey on spherical designs and algebraic combinatorics on spheres *Eur. J. Comb.* **30** 1392–425

[25] Bhamre T, Zhao Z and Singer A 2017 Mahalanobis distance for class averaging of cryo-EM images *2017 IEEE 14th Int. Symp. on Biomedical Imaging* pp 654–8

[26] Natterer F 2001 The Mathematics of Computerized Tomography *Classics in Applied Mathematics* (Philadelphia, PA: SIAM)

[27] Singer A and Shkolnisky Y 2011 Three-dimensional structure determination from common lines in cryo-EM by eigenvectors and semidefinite programming *SIAM J. Imag. Sci.* **4** 543–72

[28] Van Heel M 1987 Angular reconstitution: *a posteriori* assignment of projection directions for 3D reconstruction *Ultramicroscopy* **21** 111–23

[29] Vainshtein B K and Goncharov A B 1986 Determination of the spatial orientation of arbitrarily arranged identical particles of unknown structure from their projections *Soviet Phys. Doklady* **31** 278

[30] Coifman R R and Weiss G 1969 Representations of compact groups and spherical harmonics *L'Enseignement Mathematique* **14** 121–73

[31] Vanderberghe L and Boyd S 1996 Semidefinite programming *SIAM Rev.* **38** 49–95

[32] Dumitrescu B 2007 *Positive Trigonometric Polynomials and Signal Processing Applications* (Berlin: Springer) http://doi.org/10.1007/978-1-4020-5125-8

[33] Parrilo P A 2000 Structured Semidefinite Programs and Semialgebraic Geometry Methods in Robustness and optimization *Doctoral thesis* (California Institute of Technology)

[34] Parrilo P A, Blekherman G and Thomas R R 2013 *Semidefinite Optimization and Convex Algebraic Geometry* (Thailand: MOS-SIAM Series on Optimization)

[35] Wigner E P 1959 *Group Theory and its Application to the Quantum Mechanics of Atomic Spectra* (New York: Academic)

[36] Askey R 1975 *Orthogonal Polynomials and Special Functions* 4th edn (Philadelphia, PA: SIAM)

[37] Yershova A, Jain S, LaValle S M and Mitchell J C 2010 Generating Uniform Incremental Grids on SO(3) Using the Hopf Fibration *Int. J. Robot Res.* **29** 801–12

[38] Saunderson J, Parrilo P A and Willsky A S 2015 Semidefinite descriptions of the convex hull of rotation matrices *SIAM J. Optim.* **25** 1314–43

[39] Rodrigues O 1840 Des lois géometriques qui regissent les déplacements d' un systéme solide dans l' espace, et de la variation des coordonnées provenant de ces déplacement considérées indépendent des causes qui peuvent les produire *J. Math. Pure Appl.* **5** 380–440

[40] Zhao Z, Shkolnisky Y and Singer A 2016 Fast steerable principal component analysis *IEEE Transactions on Computational Imaging* **2** 1–12

[41] Zhao Z and Singer A 2014 Rotationally invariant image representation for viewing direction classification in cryo-EM *J. Struct. Biol.* **186** 153–66

[42] *ASPIRE—Algorithms for Single Particle Reconstruction Software Package* http://spr.math.princeton.edu/

[43] Farrow N A and Ottensmeyer E P 1992 *A posteriori* determination of relative projection directions of arbitrarily oriented macromolecules *J. Opt. Soc. Am.* A **9** 1749–60

[44] Penczek P A, Zhu J and Frank J 1996 A common-lines based method for determining orientations for $n > 3$ particle projections simultaneously *Ultramicroscopy* **63** 205–18

[45] Mallick S P, Agarwal S, Kriegman D J, Belongie S J, Carragher B and Potter C S 2006 Structure and view estimation for tomographic reconstruction: A Bayesian approach *IEEE Conf. on Computer Vision and Pattern Recognition* vol 2 pp 2253–60

[46] Singer A, Coifman R R, Sigworth F J, Chester D W and Shkolnisky Y 2010 Detecting consistent common lines in cryo-EM by voting *J. Struct. Biol.* **169** 312–22

[47] Boyd S and Vandenberghe L 2004 *Convex Optimization* (Cambridge: Cambridge University Press)

[48] Sun D, Toh K C and Yang L 2015 A convergent 3-block semiproximal alternating direction method of multipliers for conic programming with 4-type constraints *SIAM J. Optim.* **25** 882–915

[49] Prucnal P R and Saleh B E A 1981 Transformation of image-signal-dependent noise into image-signal-independent noise *Opt. Lett.* **6** 316–8

[50] Kostlee P J and Rockmore D N 2008 FFTs on the rotation group *J. Fourier Anal. Appl.* **14** 145–79

[51] Van Heel M and Schatz M 2005 Fourier shell correlation threshold criteria *J. Struct. Biol.* **151** 250–62