Describing geometry and symmetry of cryo-EM datasets using algebra

By

David Dynerman

A DISSERTATION SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY
(MATHEMATICS)

at the
UNIVERSITY OF WISCONSIN – MADISON
2015

Date of final oral examination: May 7, 2015

The dissertation is approved by the following members of the Final Oral Committee:

Professor Shamgar Gurevich, Associate Professor, Mathematics
Professor Nigel Boston, Professor, Mathematics
Professor Julie C. Mitchell, Professor, Mathematics
Professor Robert D. Nowak, Professor, Electrical & Computer Engineering
Professor Ivan Rayment, Professor, Biochemistry
Abstract

Cryo-electron microscopy (cryo-EM) is a microscopy technique used to discover the 3D structure of molecules from very noisy 2D images. This thesis presents two projects that use algebra to describe cryo-EM datasets. In the first part, we obtain an algebraic description of common lines datasets. Common lines are lines of intersection between cryo-EM images in 3D. They are an important ingredient in some 2D to 3D reconstruction algorithms, and they can be characterized by polynomial equalities and inequalities. In the second part, we show how 3D symmetries of a molecule can be detected from only 2D cryo-EM images, without performing full 3D reconstruction.
Acknowledgements

This thesis is dedicated to the many loved ones in my life, past and present.

During my time in Madison I had the good fortune to have many fantastic mentors and meet many wonderful friends. This work is as much due to them as it is to me.

I am very lucky to have worked with Shamgar who was absolutely devoted to my growth as a mathematician. Perhaps even more importantly, he worked tirelessly to instill his deep commitment to honest scientific investigation in me. I am very grateful for all that he’s done.

I was also very lucky to have studied with Marty Isaacs at the beginning of graduate school. Marty’s mentorship had a big impact on my decision to study algebra. I’m lucky to count him as a friend.

I cannot imagine the past seven years without the people that have been close to me. Andrew and Mushfeq have been the most loyal and devoted friends anyone could hope for. I hope we continue to bring joy into each other’s lives for many years. Sara’s friendship and support was very important to me throughout graduate school. Bess taught me about being kind and jumping in feet first. Erin introduced me to many wonderful things and set an inspiring example. Cecilia and Claire’s warmth and kindness transformed our apartment into a special place. Emily’s friendship turned many difficult days into happy ones.

Finally, I’d like to thank my father, who has been my biggest role model. His uncompromising passion to pursue things that are personally meaningful has made this work possible.
List of Figures

1.1 Cryo-EM produces 3D models from noisy images ........................................ 2
2.1 Cryo-EM obtains a 3D structure from noisy 2D images $I_1, \ldots, I_N$ ........ 4
2.2 Cryo-EM mathematical model ..................................................................... 5
2.3 Common line of $F_i$ and $F_j$. .................................................................... 7
2.4 Common lines data for $N = 3$ planes. ....................................................... 9
2.5 Angular reconstitution .............................................................................. 10
2.6 Raw cryo-EM image of $\beta$-galactosidase. Image by Richard Henderson, personal communication .......................................................... 11
2.7 Common lines in $P_1$, $P_2$ and $P_3$ determine a spherical triangle .......... 13
2.8 Inconsistent reconstruction from invalid common lines data .................... 17
2.9 Triangle $T(i, j, k)$ on the surface of the sphere ....................................... 18
2.10 $T(i, j, k)$, in green, shares edges with $T(i, j, m)$ .................................... 19
3.1 Env, HIV-1 envelope glycoprotein, [12] ..................................................... 36
3.2 Objects with cyclic point groups. Images by Emmanuel Levy .................. 37
3.3 Cryo-EM projection of Env ...................................................................... 38
3.4 GroEL, a molecule with dihedral point group $D_7$ [11] .............................. 39
3.5 Objects with dihedral point groups. Images by Emmanuel Levy ......... 39
3.6 Flowchart algorithm for determining the point groups of Env and GroEL .... 40
3.7 Objects exhibiting each possible point group .......................................... 42
3.8 Projections of GroEL along multiple symmetry axes ................................. 45
3.9 Rotational auto-correlation of clean Env top view Figure 3.3b ................... 51
3.10 Rotational auto-correlation of noisy Env, SNR = $\frac{1}{4}$ ............................. 52
## List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Symmetry orders appearing in point groups</td>
<td>44</td>
</tr>
<tr>
<td>3.2</td>
<td>FLOWSYM test benchmark</td>
<td>49</td>
</tr>
<tr>
<td>3.3</td>
<td>FLOWSYM Benchmark Results</td>
<td>55</td>
</tr>
</tbody>
</table>
Contents

1 Introduction 1

2 Semi-algebraic geometry of common lines 3
   2.1 Backgrounds ......................................................... 3
      2.1.1 Mathematical model for cryo-EM .......................... 4
      2.1.2 Common Lines and Reconstruction .......................... 7
      2.1.3 Angular Reconstitution ....................................... 9
      2.1.4 Noise and valid common lines data ....................... 9
   2.2 Methods .............................................................. 11
   2.3 Results & Discussion ............................................ 13
      2.3.1 Projective coordinates ...................................... 13
      2.3.2 Coordinates for Common Lines ............................. 14
      2.3.3 Necessary and sufficient conditions ....................... 15
      2.3.4 Geometry of valid common lines ......................... 21
      2.3.5 Grassmannian & Plücker coordinates ....................... 22
      2.3.6 Plücker coordinates for common lines .................... 23
   2.4 Conclusions ....................................................... 25
      2.4.1 Future Work .................................................. 25
      2.4.2 Defining Polynomials ....................................... 26
   2.5 Proofs ............................................................... 28

3 Detecting 3D molecular symmetries 35
   3.1 Introduction ....................................................... 35
      3.1.1 Detecting 3D molecular symmetries from 2D data ........ 36
      3.1.2 Detecting cyclic point groups ............................ 37
      3.1.3 Detecting non-cyclic point groups ........................ 38
      3.1.4 Our contribution and prior work .......................... 39
   3.2 Flowchart algorithm for detecting point groups ............ 41
      3.2.1 Complete list of point groups ......................... 42
      3.2.2 Cyclic Groups .............................................. 43
Chapter 1
Introduction

This thesis presents two projects on applications of algebra to cryo-electron microscopy (cryo-EM). Cryo-EM is a technique used to discover the structure of molecules, usually proteins in the context of structural biology research [21]. The goal of cryo-EM reconstruction is to create an accurate 3D model, Figure 1.1b, of a molecule given a large amount of very noisy 2D images, see Figure 1.1a.

The first project, discussed in Chapter 2, describes the shape of common lines datasets. Common lines datasets are extracted from cryo-EM images and can be used to produce the final 3D model. 3D reconstruction from common lines is well understood theoretically, but cannot be directly used in practice due to the large amounts of noise in cryo-EM images. In this chapter, we give an algebraic description of common lines datasets - they are solutions to a system of polynomial equalities and inequalities. In Section 2.4.1 we speculate on some applications of this work that may improve common-lines based reconstruction in cryo-EM.

Chapter 3 describes an algorithm, joint with Yoel Shkolnisky and Shamgar Gurevich, to detect the point group of a molecule from cryo-EM images. The point group of a molecule is the collection of all of the molecule’s symmetries, i.e., the collection of all rotations of 3D space that leave the molecule unchanged. The algorithm’s

\[\text{This is also called a semi-algebraic description}\]
correctness is due to an algebraic fact: Klein’s classification of the finite groups of rotations.
Chapter 2

Semi-algebraic geometry of common lines

2.1 Backgrounds

A basic overview of cryo-EM is presented in Figure 2.1. First, a sample is prepared by freezing many different copies of the molecule in a thin layer of ice. A stream of electrons then passes through the sample and is detected by cameras that produce $N$ noisy 2D cryo-EM images $I_1, \ldots, I_N$. The primary goal is to reconstruct the 3D structure of the molecule from the 2D images that are acquired. For a more detailed overview, see Section 1 in [17].

**Problem 2.1.1** (Reconstruction Problem: Structural Biology). *Given $N$ two-dimensional experimental cryo-EM images $I_1, \ldots, I_N$, reconstruct a three-dimensional model of the original molecule.*
2.1.1 Mathematical model for cryo-EM

We briefly describe the mathematical model for cryo-EM, following Section 0 in [6]. We work in the three dimensional space $\mathbb{R}^3$ equipped with the usual inner product.

The molecule is modeled by a function $\phi : \mathbb{R}^3 \to \mathbb{R}$ that represents its electronic density at various spatial locations, Figure 2.2a. An actual cryo-EM experiment obtains a single image of many copies of the molecule, but we instead assume that each image is a picture of the same molecule from different microscope orientations, Figure 2.2b. To model a microscope orientation we use the following concept.

Definition 2.1.2. A frame $F$ for $\mathbb{R}^3$ is an ordered orthonormal basis $(a, b, c)$ such that the determinant of the matrix $[a \, b \, c]$ is $+1$, or, equivalently, that $c = a \times b$, where $\times$ is the standard cross product on $\mathbb{R}^3$.

Remark 2.1.3. A frame $F$ for $\mathbb{R}^3$ is uniquely determined by the vectors $(a, b)$. For the rest of the paper we identify frames $(a, b, c)$ with pairs of orthonormal vectors $(a, b)$.

For us a microscope orientation is a frame $F = (a, b)$. We think of the span of the vectors $a$ and $b$ as the embedded image plane of this orientation, and the vector
c = a × b as the "viewing" direction Figure 2.2c.

A cryo-EM experiment produces \( N \) images which we denote \( I_1, \ldots, I_N \) — see Figure 2.2b. We will write \( F_i = (a_i, b_i) \) for the microscope orientation of image \( I_i \).

The embedded image plane spanned by \( a_i, b_i \) can be canonically identified with the plane \( P_i = \mathbb{R}^2 \). We think of \( P_i \) as the \textit{unembedded} image plane of \( I_i \). We model the image \( I_i \) as a real valued function on \( P_i = \mathbb{R}^2 \). The value of the image \( I_i \) at the point \((x, y)\) is the integral of \( \phi \) along a line perpendicular to the embedded image plane \( \text{span}\{a_i, b_i\} \) — see Figure 2.2d and Equation 2.1.4. This is the \textit{X-ray transform} of \( \phi \) onto the frame \( F_i \), given by

\[
I_i : P_i = \mathbb{R}^2 \to \mathbb{R}, \\
I_i(x, y) = \int_{-\infty}^{\infty} \phi(xa_i + yb_i + zc_i)dz, \quad (2.1.4)
\]
where \( c_i = a_i \times b_i \). As in [6], to solve this reconstruction problem we assume that the X-ray projections \( I_i \) and \( I_j \) of \( \phi \) from different microscope orientations \( F_i \) and \( F_j \) are different. This is equivalent to requiring the molecule \( \phi \) to admit no non-trivial symmetry as a function on \( \mathbb{R}^3 \).

In terms of this mathematical model, the goal of cryo-EM reconstruction, Problem 2.1.1, becomes to recover the function \( \phi \) from the \( N \) X-ray projections \( I_1, \ldots, I_N \). A commonly used approach for this problem is to first recover the \( N \) projection orientations \( F_1, \ldots, F_N \), see Section 0.1 of [6]. Note that the detected image \( I_i \) is a function on the plane \( P_i = \mathbb{R}^2 \), and a cryo-EM experiment does not directly provide information about the microscope orientation \( F_i \) used to compute \( I_i \).

Once the original microscope orientations are known, the unembedded image data \( I_1, \ldots, I_N \) can be placed in the original positions from where these X-ray projections were computed. Then the X-ray transform can be inverted to yield an approximation of \( \phi \). Thus, although the ultimate goal is to solve Problem 2.1.1 we instead discuss solutions to the following problem.

**Problem 2.1.5** (Reconstruction Problem: Microscope Orientations). Given \( N \) X-ray projections \( I_1, \ldots, I_N \) of a molecule \( \phi : \mathbb{R}^3 \rightarrow \mathbb{R} \), computed from the \( N \) unknown microscope orientations \( F_1, \ldots, F_N \), recover these orientations up to global rotation.

**Remark 2.1.6.** By “up to global rotation” we mean that instead of recovering the molecule \( \phi \), we instead recover the molecule \( \phi \) rotated by an element \( R \) in \( O(3) \), the group of \( 3 \times 3 \) orthogonal matrices. The matrix \( R \) may be a proper (\( \det R = +1 \)) or improper (\( \det R = -1 \)) rotation, so we expect chiral ambiguity in the reconstructed molecule.
2.1.2 Common Lines and Reconstruction

One approach for solving Problem 2.1.5 is to exploit common lines of intersection between the embedded image planes, which we now describe.

\[
\begin{align*}
\Pi_i &\quad \text{and} \quad \Pi_j, \\
F_i &\quad \text{and} \quad F_j.
\end{align*}
\]

Figure 2.3: Common line of \(F_i\) and \(F_j\).

A cryo-EM experiment produces images \(I_i\) and \(I_j\) from orientations \(F_i = (a_i, b_i)\) and \(F_j = (a_j, b_j)\). These frames define isometric embeddings \(\iota_i\) and \(\iota_j\) Figure 2.3 of the unembedded image planes \(\Pi_i\) and \(\Pi_j\) into \(\mathbb{R}^3\), given by

\[
\iota_i(x, y) = xa_i + yb_i, \quad \iota_j(x, y) = xa_j + yb_j.
\]

The images are functions on \(\Pi_i\) and \(\Pi_j\), and we know that they were obtained as X-ray projections onto the unknown embedded image planes \(\iota_i(\Pi_i)\) and \(\iota_j(\Pi_j)\) Figure 2.3b. We assume that each embedded image plane \(\iota_i(\Pi_i)\) is distinct, and, further, that each pair of such planes intersects in a distinct line. Such a configuration of microscope orientations is called generic. The microscope orientations will be generic if they are sampled uniformly from the space of all frames, as, for example, assumed in the eigenvector relaxation algorithm developed in Section 3 of [17].

The embedded image planes \(\iota_i(\Pi_i)\) and \(\iota_j(\Pi_j)\) intersect in a line \(L\) Figure 2.3b,
and this line corresponds to the unembedded lines $\ell_{ij} \subset P_i$ and $\ell_{ji} \subset P_j$. Since these unembedded lines both came from $L \subset \mathbb{R}^3$, we have a natural choice $\psi_{ij}: \ell_{ij} \rightarrow \ell_{ji}$ of one of the two possible isometries between $\ell_{ij}$ and $\ell_{ji}$. Proceeding in this fashion, the $N$ microscope orientations $F_1, \ldots, F_N$ produce $\binom{N}{2} = N(N-1)/2$ common line pairs $\{(\ell_{ij}, \ell_{ji}, \psi_{ij})\}$. This is the common lines data realized by the frames $F_1, \ldots, F_N$. It will be useful for us to distinguish such common lines data obtained from frames.

**Definition 2.1.8.** A common line pair for $P_i$ and $P_j$ is a pair of lines $\ell_{ij} \subset P_i$ and $\ell_{ji} \subset P_j$, together with a choice of isometry $\psi_{ij}: \ell_{ij} \rightarrow \ell_{ji}$. A collection of common line pairs $\{(\ell_{ij}, \ell_{ji}, \psi_{ij})\}$, for every $P_i$ and $P_j$, is common lines data for $P_1, \ldots, P_N$. We say common lines data is valid if it is realized by some generic frames $F_1, \ldots, F_N$.

Despite the fact that common lines data is information in the unembedded planes $P_i$, it is a fact that, when $N \geq 3$, valid common lines data determines its realizing frames, up to global rotation. Further, algorithms have long been known (e.g. Section 2.1 of [7]) that recover a set of realizing frames from valid common lines data.

This is relevant to cryo-EM reconstruction, because although the microscope orientations are unknown, it is possible to detect the common lines data the orientations realize from the images $I_1, \ldots, I_N$ [3]. Thus we have the following common lines approach for the cryo-EM reconstruction problem, Problem 2.1.5. We first detect the common lines data realized by the unknown microscope orientations. Next, from the valid common lines data we reconstruct a set of realizing frames. Since valid common lines data determines its realizing frames up to global rotation, the reconstructed frames are related to the original microscope orientations by a global rotation, and so in principle one has solved the reconstruction problem.
2.1.3 Angular Reconstitution

In this section we describe the *angular reconstitution algorithm*, due to van Heel \cite{vanHeel7}, and also independently Vainshtein and Goncharov \cite{Vainshtein9}, which recovers a set of realizing frames from valid common lines data.

Our input is valid common lines data \(\{ (\ell_{ij}, \ell_{ji}, \psi_{ij}) \} \) for \( P_1, \ldots, P_N \), Figure 2.4. Note that recovering a frame \( F_i \) is equivalent to recovering the embedding \( \iota_i \) of \( P_i \), which will be easier to visualize. Since we are only reconstructing up to global rotation, the first step is to embed \( P_1 \) in an arbitrary position in \( \mathbb{R}^3 \), Figure 2.5a. Next, we use the isometry \( \psi_{12} \) between \( \ell_{12} \) and \( \ell_{21} \) to dock \( P_2 \) to \( \iota_1(P_1) \), Figure 2.5b. This docking is ambiguous, see Figure 2.5c, since we are free to rotate \( \iota_2(P_2) \) about its line of intersection with \( \iota_1(P_1) \). We resolve this ambiguity by docking \( P_3 \) with \( \iota_1(P_1) \) and matching up \( \ell_{23} \) and \( \ell_{32} \) in \( \iota_2(P_2) \) and \( \iota_3(P_3) \), Figure 2.5d. We continue in this fashion, docking each subsequent plane \( P_i \) with \( \iota_1(P_1) \) and resolving the rotational ambiguity by comparing against the remaining frames.

2.1.4 Noise and valid common lines data

We discussed in Section 2.1.2 that valid common lines data determines its realizing frames up to global rotation. Common lines based approaches for cryo-EM reconstruction, Problem 2.1.5, assume that we can accurately detect the valid common
Figure 2.5: Angular reconstitution.

Misdetected common lines pose a problem because they lead to inconsistencies when attempting to recover realizing frames. For example, in Figure 2.5, we resolved the ambiguity of $\iota_2(P_2)$ by docking $P_3$ to $\iota_1(P_1)$ and using the common lines $l_{23}$ and $l_{32}$ Figure 2.5c. However, we could have equally well resolved the ambiguity of $\iota_2(P_2)$ by docking $P_4$ and using the common lines $l_{24}$ and $l_{42}$. Thus, if we, for example, incorrectly identify the common lines in $P_4$ we will have two contradictory embeddings $\iota_2(P_2)$ with no obvious way of determining which is correct.

More generally, the angular reconstitution algorithm makes many choices: for example which plane to begin reconstruction with, and how to resolve docking ambiguities. The final reconstructed frames depend on all these choices. By definition valid common lines data is precisely the data which has a single consistent (up
Figure 2.6: Raw cryo-EM image of $\beta$-galactosidase. Image by Richard Henderson, personal communication.

to global rotation) set of realizing frames. The development of common lines reconstruction algorithms that are robust to this kind of error is an active area of research.

2.2 Methods

We wish to understand the set $C_N$ of all valid common lines data for $N$ planes $P_1, \ldots, P_N$. First, we derive necessary and sufficient conditions for common lines data to be valid. These conditions are polynomial equations and inequalities, which means that $C_N$ is a semi-algebraic set, and allows us to study $C_N$ as a geometric space. In particular, we compute the dimension of $C_N$, and show that there is a bijection between $C_N$ and the space of generic frames, up to global rotation.

Main Theorem. The set $C_N$ of all valid common lines data for $N$ frames is a $3N - 3$ dimensional semi-algebraic subset of the $2\binom{N}{2}$ dimensional space of all common lines data, and is in bijection with the space of $N$ generic frames modulo $O(3)$. The defining equations for $C_N$ are given by $\binom{N}{3}$ polynomial inequalities arising from the spherical triangle inequalities and $6\binom{N}{4}$ polynomial equalities arising from the
The spherical law of cosines.

The meaning of this theorem is as follows. As we discussed in Section 2.1.2, one way to obtain valid common lines data is from the embedded frames $F_1, \ldots, F_N$. The theorem provides an intrinsic definition of this valid common lines data, namely, the defining polynomials for $C_N$. This is a definition for valid common lines only in terms of the data $\{(l_{ij}, l_{ji}, \psi_{ij})\}$ on unembedded planes $P_1, \ldots, P_N$, and without reference to any embedded frames $F_1, \ldots, F_N$.

We briefly describe the idea behind our proofs. Suppose we have valid common lines data

$$\{(\ell_{12}, \ell_{21}, \psi_{12}), (\ell_{13}, \ell_{31}, \psi_{13}), (\ell_{23}, \ell_{32}, \psi_{23})\}. \tag{2.2.1}$$

The angles between these unembedded common lines determine a triangle on the unit sphere in $\mathbb{R}^3$, see Figure 2.7, and so the angles $\alpha$ between $\ell_{12}$ and $\ell_{13}$, $\beta$ between $\ell_{21}$ and $\ell_{23}$ and $\gamma$ between $\ell_{31}$ and $\ell_{32}$ must satisfy the spherical triangle inequalities. These inequalities are analogs of the plane triangle inequality, i.e. necessary and sufficient conditions for a spherical triangle to exist with the specified edge lengths. In other words, a necessary and sufficient condition for common lines data to be valid for $N = 3$ is that it satisfy the spherical triangle inequality, a fact already observed both by the cryo-EM, see pp. 198-199 \[20\], and mathematics, see Equations 11 & 12 \[16\], communities.

We prove our results for $N > 3$ by similarly appealing to spherical trigonometry. Specifically, given common lines data $\{\ell_{ij}, \ell_{ji}, \psi_{ij}\}$ for $N$ planes, we require that for each triple $1 \leq i < j < k \leq N$ the common lines data $(\ell_{ij}, \ell_{ji}, \psi_{ij}), (\ell_{ik}, \ell_{ki}, \psi_{ik})$ and $(\ell_{jk}, \ell_{kj}, \psi_{jk})$ satisfy the spherical triangle inequalities. Now, reducing to the $N = 3$ case gives us realizing embeddings $\iota_i$, $\iota_j$, $\iota_k$ for each triple $(i, j, k)$ of indices. To reconstruct a collection of $N$ consistent frames, all these triple reconstructions must
be compatible. We show that this compatibility condition is a polynomial condition arising from the spherical law of cosines. These defining equations are given by polynomials which are explicitly derived, and listed in Section 2.4.2.

2.3 Results & Discussion

We proceed to describe in detail the results in our Main Theorem. We will derive necessary and sufficient conditions for common lines data for $N \geq 3$ to be valid. These will be explicit polynomial equations and inequalities only in the unembedded information $\{(l_{ij}, l_{ji}, \psi_{ij})\}$, and will provide an intrinsic definition for valid common lines without reference to the frames $F_1, \ldots, F_N$. We defer all proofs to Section 2.5.

2.3.1 Projective coordinates

To obtain defining equations for $C_N$ it will be convenient for us to work with projective coordinates, which we briefly review. Suppose $V$ is a vector space and $\ell$ is a line in $V$ through the origin. We can represent $\ell$ by choosing any non-zero vector $v \in \ell$. In other words, lines can be identified with equivalence classes of vectors.
under scaling. We denote the equivalence class of a vector \( v \) by \([v]\), and by definition \([v] = [w]\) if and only \( v = \lambda w \), for some \( \lambda \neq 0 \). The space of all lines through the origin in \( V \) is the projective space \( \mathbb{P}(V) \). If \( V = U \times W \) and \((u, w) \in V\), then we write \([u : w]\) for the corresponding equivalence class in \( \mathbb{P}(U \times W) \).

### 2.3.2 Coordinates for Common Lines

Suppose now that \((\ell_{ij}, \ell_{ji}, \psi_{ij})\) is a common line pair for \( P_i \) and \( P_j \). Choose a non-zero vector \( v_{ij} = (x_{ij}, y_{ij}) \) on the line \( \ell_{ij} \subset P_i \), and consider the pair \((v_{ij}, \psi_{ij}(v_{ij}))\) \( \in P_i \times P_j \). Note that different choices of a vector along \( \ell_{ij} \) will simply scale \((v_{ij}, \psi_{ij}(v_{ij}))\) by a non-zero multiple, so the projective pair \([v_{ij} : \psi_{ij}(v_{ij})]\) in \( \mathbb{P}(P_i \times P_j) \) is uniquely determined by \((\ell_{ij}, \ell_{ji}, \psi_{ij})\).

Conversely, if \([v_{ij} : v_{ji}]\) \( \in \mathbb{P}(P_i \times P_j) \) satisfies \( \|v_{ij}\|^2 = \|v_{ji}\|^2 \), then choosing representatives \((v_{ij}, v_{ji})\), we obtain a common line pair \((\text{span}\{v_{ij}\}, \text{span}\{v_{ji}\}, \psi_{ij})\), where \( \psi_{ij} \) is the unique isometry that sends \( v_{ij} \mapsto v_{ji} \). Note that we obtain the same common line pair regardless of which representing vectors we choose.

Thus, from now on we identify common line pairs with elements \([v_{ij} : v_{ji}]\) \( \in \mathbb{P}(P_i \times P_j) \) satisfying \( \|v_{ij}\|^2 = \|v_{ji}\|^2 \). We also apply this identification to common lines data:

**Remark 2.3.1.** We identify common lines data for \( P_1, \ldots, P_N \) with collections

\[
([v_{ij} : v_{ji}]) \in \prod_{1 \leq i < j \leq N} \mathbb{P}(P_i \times P_j) = \mathbb{P}^3(\mathbb{P}^3)^{(N\choose2)}
\]

that satisfy \( \|v_{ij}\|^2 = \|v_{ji}\|^2 \) for all pairs.

In coordinates we say that the frames \( F_i \) and \( F_j \) realize the common line pair \([v_{ij} : v_{ji}]\) if the associated embeddings, Equation 2.1.7, bring together this common line pair, i.e. for any choice of representative \((v_{ij}, v_{ji})\), we have
\[ \iota_i(v_{ij}) = \iota_j(v_{ji}). \]

By definition valid common lines data is a collection \([(v_{ij} : v_{ji})]\) of common lines data for which there exist frames \(F_1, \ldots, F_N\) such that for all \(1 \leq i < j \leq N\), the frames \(F_i\) and \(F_j\) realize \([v_{ij} : v_{ji}]\).

### 2.3.3 Necessary and sufficient conditions

In this section we derive equations and inequalities that are necessary and sufficient for common lines data \([(v_{ij} : v_{ji})]\) to be valid. We first discuss necessary conditions. Recall from Section 2.2 that for any triple of indices \(i, j, k\) the angles between the common line pairs \([v_{ij} : v_{ji}], [v_{ik} : v_{ki}]\) and \([v_{jk} : v_{kj}]\) determine a spherical triangle Figure 2.7 and so these angles must satisfy the spherical triangle inequalities. The spherical triangle inequalities state that a non-degenerate spherical triangle of edge lengths \(\alpha, \beta\) and \(\gamma\), all in \((0, \pi)\), exists if and only if

\begin{align*}
\beta + \gamma & > \alpha, \\
\alpha + \gamma & > \beta, \\
\alpha + \beta & > \gamma, \\
\alpha + \beta + \gamma & < 2\pi.
\end{align*}

(2.3.2)

**Remark 2.3.3.** Fix common lines data \([(v_{ij} : v_{ji})] \in (\mathbb{P}^3)^N\) and a triple of indices \((i, j, k)\). If we choose representatives \((v_{ij}, v_{ji}), (v_{ik}, v_{ki})\) and \((v_{jk}, v_{kj})\), we can write

\[ \alpha_{ijk} = \cos^{-1}\left( \frac{v_{ij} \cdot v_{ik}}{\|v_{ij}\|\|v_{ik}\|} \right), \quad \beta_{ijk} = \cos^{-1}\left( \frac{v_{ji} \cdot v_{jk}}{\|v_{ji}\|\|v_{jk}\|} \right), \quad \gamma_{ijk} = \cos^{-1}\left( \frac{v_{ki} \cdot v_{kj}}{\|v_{ki}\|\|v_{kj}\|} \right). \]
The angles $\alpha_{ijk}, \beta_{ijk}$ and $\gamma_{ijk}$ depend on the representatives we have chosen, however whether or not the spherical triangle inequalities, Equation 2.3.2 are satisfied by $\alpha_{ijk}, \beta_{ijk}, \gamma_{ijk}$ is independent of this choice. Thus, we can make the following definition:

**Definition 2.3.4.** Fix common lines data $([v_{ij} : v_{ji}]) \in (\mathbb{P}^3)^{(N)}$ and a triple of indices $(i, j, k)$. We say $(i, j, k)$ satisfies the triangle inequalities if, for any choice of representatives $(v_{ij}, v_{ji}), (v_{ik}, v_{ki})$ and $(v_{jk}, v_{kj})$, the angles $\alpha_{ijk}, \beta_{ijk}, \gamma_{ijk}$ satisfy Equation 2.3.2.

This definition allows us to state our first result.

**Proposition 2.3.5.** Fix common lines data $([v_{ij} : v_{ji}]) \in (\mathbb{P}^3)^{(N)}$, and suppose that the triple $(i, j, k)$ satisfies the spherical triangle inequalities. Then there exist generic frames $F_i, F_j, F_k$ that realize the common line pairs $[v_{ij} : v_{ji}], [v_{ik} : v_{ki}]$ and $[v_{jk} : v_{kj}]$. Moreover if $G_i, G_j, G_k$ are another set of frames that realize these same pairs, then there exists an isometry in $O(3)$ that maps $(F_i, F_j, F_k) \mapsto (G_i, G_j, G_k)$.

For a proof of this proposition, see Section 2.5. When $([v_{ij} : v_{ji}])$ is fixed, and the common lines $[v_{ij} : v_{ji}], [v_{ik} : v_{ki}]$ and $[v_{jk} : v_{kj}]$ are realized by $F_i, F_j$ and $F_k$, we will say that these frames realize the triple $(i, j, k)$.

This proposition is a necessary and sufficient condition for realizing frames to exist for a triple $(i, j, k)$, and so we have recovered necessary and sufficient conditions for $N = 3$. For $N > 3$, this proposition states that each triple of indices $(i, j, k)$ must satisfy the spherical triangle inequality, but this condition is no longer sufficient.

**Example 2.3.6.** Consider the common lines data for $P_1, P_2, P_3, P_4$ given by

$$(v_{12}, v_{13}, v_{14}) = (v_{21}, v_{23}, v_{24}) = (v_{31}, v_{32}, v_{34}) = (v_{41}, v_{42}, v_{43}) = ([1, 0]^T, [\sqrt{2}/2, \sqrt{2}/2]^T, [0, 1]^T).$$
The angles between these common lines are given by

\[
\begin{aligned}
(\alpha_{123}, \beta_{123}, \gamma_{123}) &= \left(\frac{\pi}{4}, \frac{\pi}{4}, \frac{\pi}{4}\right), \\
(\alpha_{124}, \beta_{124}, \gamma_{124}) &= \left(\frac{\pi}{2}, \frac{\pi}{2}, \frac{\pi}{4}\right), \\
(\alpha_{134}, \beta_{134}, \gamma_{134}) &= \left(\frac{\pi}{4}, \frac{\pi}{2}, \frac{\pi}{2}\right), \\
(\alpha_{234}, \beta_{234}, \gamma_{234}) &= \left(\frac{\pi}{4}, \frac{\pi}{4}, \frac{\pi}{4}\right).
\end{aligned}
\]

Observe that each of these triples satisfies the spherical triangle inequality. However, this data cannot be realized by frames \(F_1, F_2, F_3\) and \(F_4\), and so this common line data is not valid. To see why, suppose such frames existed and, for each pair \(i, j\), set \(\Lambda_{ij} = \iota_i(v_{ij}) = \iota_j(v_{ji})\). The points \(\Lambda_{12}, \Lambda_{13}, \Lambda_{23}\) determine a spherical triangle with edge lengths \((\alpha_{123}, \beta_{123}, \gamma_{123})\) Figure 2.8a, and the angle of this spherical triangle at the vertex between edges \(\alpha_{123}\) and \(\beta_{123}\) is exactly the angle \(\theta_{12}\) between the planes \(\iota_1(P_1)\) and \(\iota_2(P_2)\). From the spherical law of cosines, we can compute this angle:

\[
\cos \theta_{12} = \frac{\cos \gamma_{123} - \cos \alpha_{123} \cos \beta_{123}}{\sin \alpha_{123} \sin \beta_{123}} = \sqrt{2} - 1.
\]

Similarly, the points \(\Lambda_{12}, \Lambda_{14}\) and \(\Lambda_{24}\) determine a spherical triangle with edge lengths \((\alpha_{124}, \beta_{124}, \gamma_{124})\), Figure 2.8b, and the angle of this triangle between edges
\(\alpha_{124}\) and \(\beta_{124}\) is again the angle \(\theta_{12}\) between the planes \(\iota_1(P_1)\) and \(\iota_2(P_2)\). However, in this triangle we have \(\cos \theta_{12} = \sqrt{2}/2\), which is a contradiction.

\[\begin{array}{c}
\Lambda_{i,j} \\
\Lambda_{i,k} \\
\Lambda_{j,k} \end{array} \] 

\[\begin{array}{c}
\iota_i(P_i) \\
T(i, j, k) \\
\iota_j(P_j) \\
\iota_k(P_k) \end{array} \]

Figure 2.9: Triangle \(T(i, j, k)\) on the surface of the sphere.

We now provide an explanation for why the contradiction in Example 2.3.6 arose that will lead us to necessary and sufficient conditions for reconstruction when \(N > 3\).

Suppose the frames \(F_1, \ldots, F_N\) realize the common lines data \([(v_{ij} : v_{ji})] \in (\mathbb{P}^3)^{\binom{N}{2}}\), and choose unit vector representatives \((v_{ij}, v_{ji})\) for all the common line pairs. If we consider the intersection of the embedded planes \(\iota_i(P_i)\) with the unit sphere in \(\mathbb{R}^3\), we obtain \(N\) great circles. Each pair of these great circles has a distinguished point of intersection \(\iota_i(v_{ij}) = \iota_j(v_{ji})\) which we denote by \(\Lambda_{ij}\). Denote by \(T(i, j, k)\) the triangle obtained by taking \(\Lambda_{ij}, \Lambda_{ik}\) and \(\Lambda_{jk}\) as vertices Figure 2.9.

Consider the second triangle \(T(i, j, m)\) Figure 2.10. The two triangles \(T(i, j, k)\) and \(T(i, j, m)\) share a vertex, \(\Lambda_{ij}\), and the edges of both triangles at this vertex lie in \(\iota_i(P_i)\) and \(\iota_j(P_j)\). It follows that the angle \(Z\) in \(T(i, j, k)\) and \(Z'\) in \(T(i, j, m)\) at this common vertex must be compatible: the angles are either the same, Figure 2.10a, or supplementary, Figure 2.10b, depending on the arrangement of the vertices. We can express this requirement in terms of the common lines data by using the spherical law of cosines:
(a) Angle $Z$ in $T(i, j, k)$ equals $Z'$ in $T(i, j, m)$.

(b) Angle $Z$ in $T(i, j, k)$ is supplementary to $Z'$ in $T(i, j, m)$.

Figure 2.10: $T(i, j, k)$, in green, shares edges with $T(i, j, m)$.

$$
(\cos \gamma_{123} - \cos \alpha_{123} \cos \beta_{123}) \sin \alpha_{124} \sin \beta_{124} = \\
\sigma(\cos \gamma_{124} - \cos \alpha_{124} \cos \beta_{124}) \sin \alpha_{123} \sin \beta_{123},
$$

(2.3.7)

where $\sigma$ determines whether $Z = Z'$ or $Z = \pi - Z'$. In this light, the contradiction in Example 2.3.6 arose because the angles at $\Lambda_{12}$ in $T(1, 2, 3)$ and $T(1, 2, 4)$ were not compatible.

**Remark 2.3.8.** Fix common lines data $([v_{ij} : v_{ji}] \in (\mathbb{P}^3)^{(N)}_{2} \setminus [v_{ij}])$, and two triples $(i, j, k)$ and $(i, j, m)$ that agree in two indices. If we choose representatives $(v_{ij}, v_{ji})$, $(v_{ik}, v_{ki})$, $(v_{jk}, v_{kj})$, $(v_{im}, v_{mi})$ and $(v_{jm}, v_{mj})$ for these common lines, the necessary
angle equality, Equation 2.3.7 described above is

\[
L_{ijk,ijm} = 
\left( (v_{ij} \cdot v_{ij})(v_{ki} \cdot v_{kj}) - (v_{ij} \cdot v_{ik})(v_{ji} \cdot v_{jk}) \right) \det [v_{ij}, v_{im}] \det [v_{ji}, v_{jm}] - 
\sigma \left( (v_{ij} \cdot v_{ij})(v_{mi} \cdot v_{mj}) - (v_{ij} \cdot v_{im})(v_{ji} \cdot v_{jm}) \right) \det [v_{ij}, v_{ik}] \det [v_{ji}, v_{jk}],
\]

where

\[
\sigma = \text{sign}(\det [v_{ij}, v_{ik}] \det [v_{ij}, v_{im}] \det [v_{ji}, v_{jk}] \det [v_{ji}, v_{jm}]).
\]

Equation 2.3.9

Whether or not \( L_{ijk,ijm} = 0 \) is independent of the representatives we choose, so we can make the following definition.

**Definition 2.3.10.** Fix common lines data \((v_{ij} : v_{ji}) \in (\mathbb{P}^3)^{\binom{N}{2}}\) and two triples \((i,j,k)\) and \((i,j,m)\) that agree in two indices. We say \((i,j,k)\) and \((i,j,m)\) satisfy the spherical law of cosines compatibility if \( L_{ijk,ijm} = 0 \).

The spherical law of cosines compatibility is necessary for common lines data to be valid, and we will see it is sufficient as well. We first show that if this law of cosines compatibility between \((i,j,k)\) and \((i,j,m)\) is satisfied, then we can glue together realizing frames for these triples in a compatible fashion.

**Lemma 2.3.11.** Fix common lines data \((v_{ij} : v_{ji}) \in (\mathbb{P}^3)^{\binom{N}{2}}\), and suppose that the triples \((i,j,k)\) and \((i,j,m)\) satisfy the spherical law of cosines compatibility. Then, if \(F_i, F_j, F_k\) are any realizing frames for \((i,j,k)\), and \(G_i, G_j, G_m\) are any realizing frames for \((i,j,m)\), then there exists a unique isometry in \(O(3)\) that sends \(F_i \mapsto G_i\) and \(F_j \mapsto G_j\).

For a proof, see Section 2.5.

We now can show that the law of cosines compatibility is sufficient for reconstruction.
Theorem 2.3.12. Fix common lines data \((v_{ij} : v_{ji}) \in (\mathbb{P}^3)^\binom{N}{2}\), and suppose that every triple \((i, j, k)\) satisfies the spherical triangle inequality, and, further, that every pair of triples \((i, j, k)\), \((i, j, m)\) that agree in two indices satisfies the spherical law of cosines compatibility. Then there exist generic frames \(F_1, \ldots, F_N\), unique up to isometry in \(O(3)\), realizing \((v_{ij} : v_{ji})\).

For a proof, see Section 2.5.

2.3.4 Geometry of valid common lines

We now use the necessary and sufficient conditions derived above to deduce some geometric properties about the set \(C_N\) of all valid common lines. The main result in this section is that \(C_N\) is in bijection with the space of generic frames, up to global rotation. In particular, this implies that the dimension of \(C_N\) is \(3N - 3\).

We first explicitly describe how to obtain valid common lines from a set of generic realizing frames \(F_1, \ldots, F_N\), as in Section 2.1.2. For each pair \(i, j\), choose a vector \(\Lambda_{ij}\) in the one dimensional vector space \(\iota_i(P_i) \cap \iota_j(P_j)\). Since \(\mathbb{R}^3\) has the canonical structure of an inner product space, we have corresponding orthogonal projections \(\iota_i^T : \mathbb{R}^3 \to P_i\) and \(\iota_j^T : \mathbb{R}^3 \to P_j\). Consider the vectors

\[(v_{ij}, v_{ji}) = (\iota_i^T(\Lambda_{ij}), \iota_j^T(\Lambda_{ij})) \in P_i \times P_j.\]

By construction the pair \([v_{ij} : v_{ji}] = [x_{ij} : y_{ij} : x_{ji} : y_{ji}]\) is a common line pair realized by the frames \(F_i\) and \(F_j\). In coordinates, we have

\[x_{ij}a_i + y_{ij}b_i = \Lambda_{ij} = x_{ji}a_j + y_{ji}b_j.\] (2.3.13)

Repeating this process for all pairs \(1 \leq i < j \leq N\), we obtain valid common lines data \((v_{ij} : v_{ji}) \in C_N\) that is realized by \(F_1, \ldots, F_N\). This algorithmically gives a
map $G \to C_N$, where $G$ is the subset of $N$ generic frames in $\mathcal{F}^N$. It will be useful to express this function via explicit polynomial mappings. We first describe a set of coordinates on the Grassmannian $\text{Gr}(3, 2N)$, whose points are the three dimensional subspaces of $\mathbb{R}^{2N}$.

2.3.5 Grassmannian & Plücker coordinates

If $W \subset \mathbb{R}^{2N}$ is a three dimensional subspace of $\mathbb{R}^{2N}$, and we choose a basis $w_1, w_2, w_3 \in \mathbb{R}^{2N}$ for $W$, we can represent the point in $\text{Gr}(3, 2N)$ corresponding to $W$ by the vector of all $3 \times 3$ minors of the $3 \times 2N$ matrix

$$[w_1, w_2, w_3]^T.$$ 

These minors are the Plücker coordinates of the subspace $W$. If we choose a different basis for $W$, the vector of $3 \times 3$ minors will only change by a non-zero scalar. Since Plücker coordinates are only defined up to scaling, we interpret the Grassmannian $\text{Gr}(3, 2N)$ as a subvariety of the projective space $\mathbb{P}^{(2N)-1}$.

Given a collection of $N$ frames $F_1, \ldots, F_N$, we can form the $3 \times 2N$ matrix

$$F_\bullet = [F_1 \ldots F_N] = [a_1, b_1, \ldots, a_N, b_N].$$

We consider the rational map $\rho : \mathcal{F}^N \to \text{Gr}(3, 2N)$ that takes a collection of frames $F_1, \ldots, F_N$ to the Plücker coordinates of $F_\bullet$. A rational map is a map that is defined almost everywhere in the domain. In this case, $\rho$ is not defined if the rank of $F_\bullet$ is $\leq 2$, since, in this case, the rows of $F_\bullet$ do not determine a three dimensional subspace of $\mathbb{R}^{2N}$. 
2.3.6 Plücker coordinates for common lines

As described above, given a pair of frames $F_i, F_j$ for $i < j$, we can compute the associated common line pair $[v_{ij} : v_{ji}]$ by choosing any vector $\Lambda_{ij}$ in $\iota_i(P_i) \cap \iota_j(P_j)$.

In particular, we can choose $\Lambda_{ij} = (a_i \times b_i) \times (a_j \times b_j)$, where $\times$ is the standard vector cross product on $\mathbb{R}^3$. Then, the following identity from vector algebra, called the vector quadruple product, expresses $\Lambda_{ij}$ in terms of the frames $F_i$ and $F_j$:

$$\det [a_j, b_j, a_i] b_i - \det [a_j, b_j, b_i] a_i = (a_i \times b_i) \times (a_j \times b_j)$$
$$= \det [a_i, b_i, b_j] a_j - \det [a_i, b_i, a_j] b_j.$$

Comparing this with Equation 2.3.13, we see that the coordinates of the common line pair $[v_{ij} : v_{ji}]$ are given by determinants of certain $3 \times 3$ matrices. Explicitly, we have

$$v_{ij} = \begin{bmatrix} -\det [a_j, b_j, b_i] \\ \det [a_j, b_j, a_i] \end{bmatrix}, \quad v_{ji} = \begin{bmatrix} \det [a_i, b_i, b_j] \\ -\det [a_i, b_i, a_j] \end{bmatrix}.$$

Observe that these $3 \times 3$ determinants are certain $3 \times 3$ minors of the matrix $F_i$. The minors that appear are those that belong to only two frames $F_i$ and $F_j$: in other words, minors that choose any three of $\{a_i, b_i, a_j, b_j\}$ for columns. The minors not appearing as coordinates of a common line pair are those that choose three columns from three distinct frames:

$$\det \{[a_i, b_i], [a_j, b_j], [a_k, b_k]\}. \tag{2.3.14}$$

Thus, the coordinates on the Grassmannian $\text{Gr}(3, 2N)$ are the common line coordinates, together with these “bad” minors Equation 2.3.14. If we consider the
projection where we discard the “bad” minors, we obtain the rational map

\[ \text{Gr}(3, 2N) \rightarrow \prod_{1 \leq i < j \leq N} \mathbb{P}(P_i \times P_j) = (\mathbb{P}^3)^{\binom{N}{2}}. \]

Explicitly, for \( i < j \), this projection maps

\[ \ldots : - \det [a_j, b_j, b_i] : \det [a_j, b_j, a_i] : \det [a_i, b_i, a_j] : - \det [a_i, b_i, a_j] : \ldots \]

\[ \mapsto [v_{ij} : v_{ji}]. \]

Note that this rational map is not defined whenever the four \( 3 \times 3 \) minors appearing in the common line pair \( [v_{ij} : v_{ji}] \) simultaneously vanish. This cannot happen with generic frames, so this projection is an actual map when restricted to \( \rho(G) \subset \text{Gr}(3, 2N) \). The image of this map is the set of valid common lines \( C_N \), and the map is in fact a bijection.

**Theorem 2.3.15.** The restriction \( \pi \) of the projection \( \text{Gr}(3, 2N) \rightarrow (\mathbb{P}^3)^{\binom{N}{2}} \) to \( \rho(G) \subset \text{Gr}(3, 2N) \) is a bijection onto \( C_N \).

For a proof, see Section 2.5.

As we discussed above, the point \( \rho(F_\bullet) \in \text{Gr}(3, 2N) \) only determines the row space of the matrix \( F_\bullet = [F_1, \ldots, F_N] \). A different basis for this row space is given by multiplying \( F_\bullet \) on the left by a matrix \( A \) in \( \text{O}(3) \), or, equivalently, by the following action

\[ A \cdot (F_1, \ldots, F_N) = (AF_1, \ldots, AF_N). \]

This is the *diagonal action* of \( \text{O}(3) \) on the space of frames \( \mathcal{F}^N \). We observe that this \( \text{O}(3) \) action is the only ambiguity between the space of frames and the Plücker
embedding of these frames in $\text{Gr}(3, 2N)$. Since common lines data corresponds to points in $\text{Gr}(3, 2N)$, we have recovered the fact that common lines data only determines its realizing frames up to $O(3)$.

**Corollary 2.3.16.** The dimension of $C_N$ as a semi-algebraic set is $3N - 3$.

For a proof, see Section 2.5.

### 2.4 Conclusions

The polynomial equations defining $C_N$ provide the intrinsic definition for valid common lines we set out to find. We briefly discuss potential applications.

#### 2.4.1 Future Work

Thinking of valid common lines data in geometric terms provides some insight about inconsistencies during reconstruction due to noise. The space of all common lines data has dimension $N(N - 1)$, and, since valid common lines are in bijection with the space of $N$ frames up to global rotation, we have that the dimension of $C_N$ is $3N - 3$. Since $C_N$ is a space of small dimension in the ambient space, it follows that the reconstruction inconsistencies described in Section 2.1.4 are guaranteed to occur. In effect the most basic version of the angular reconstitution algorithm reconstructs the microscope orientations $F_1, \ldots, F_N$ using only $2N - 3$ out of the $\binom{N}{2}$ common line pairs, and arbitrarily ignores inconsistencies within these pairs. The set $C_N$ is precisely the set of common lines data for which this algorithm will produce the same output regardless of which common line pairs are used, but as described above we do not expect experimental data to lie in $C_N$.

Developing common lines reconstruction algorithms that are robust to noise is an active area of research. We are interested in exploring a geometric approach
to noise reduction, which we briefly describe. In principle noisy experimental data
\{(l'_{ij}, l'_{ji}, \psi'_{ij})\} that lies outside of \(C_N\) “came from” some noiseless valid common lines
data in \(C_N\). Since the set \(C_N\) is the set of solutions of a system of polynomials,
it is theoretically possible to project noisy common lines to the set of noiseless
common lines \(C_N\) via constrained polynomial optimization. We hope to develop
effective projection algorithms along these lines to reduce the impact of noise in
reconstruction.

In Section 2.3 we obtain defining polynomials for valid common lines data by ap-
pealing to spherical geometry. It is also possible to interpret valid common lines in
terms of Gram matrices, as in [16] for the case \(N = 3\). With this interpretation, for
\(N > 3\) one can attempt to find defining polynomials by eliminating certain variables
from the defining equations of low rank Gram matrices. The algebraic set corre-
spending to this elimination is the quotient of \(C_N\) by the natural action of \(SO(2)^N\)
in each image plane. We have not yet been able to solve this elimination problem us-
ning direct approaches available in the computational algebra software Macaulay 2 [5].
We are interested in further studying these related defining polynomials, since they
suggest the possibility of applying matrix completion techniques to the denoising
projection described above.

2.4.2 Defining Polynomials

In Section 2.3.3 we derived the defining equations for \(C_N\) in terms of spherical
geometry. For the benefit of the reader we now explicitly describe these conditions
as multi-homogeneous polynomials in the variables \([v_{ij} : v_{ji}]\).

Suppose \((v_{ij} : v_{ji}) \in (\mathbb{P}^3)^{N\choose 2}\) is fixed, and that \(\|v_{ij}\|^2 = \|v_{ji}\|^2\) for all \(1 \leq
i < j \leq N\). The spherical triangle inequalities for the common line pairs \([v_{ij} : v_{ji}],\n[v_{ik} : v_{ki}]\) and \([v_{jk} : v_{kj}]\) described in Equation 2.3.2 are equivalent, see Equation 11
To express the spherical law of cosines compatibilities $L_{ijk,ijm}$, Equation 2.3.9, set

$$a = (\|v_{ij}\|^2(v_{ki} \cdot v_{kj}) - (v_{ij} \cdot v_{ik})(v_{ji} \cdot v_{jk})),$$

$$b = (\|v_{ij}\|^2(v_{mi} \cdot v_{mj}) - (v_{ij} \cdot v_{im})(v_{ji} \cdot v_{jm})),$$

$$d_1 = \det[v_{ij}, v_{im}] \det[v_{ji}, v_{jm}],$$

$$d_2 = \det[v_{ij}, v_{ik}] \det[v_{ji}, v_{jk}].$$

Then $L_{ijk,ijm} = 0$ if and only if

$$a^2 d_1^2 - 2d_1 d_2 ab + b^2 d_2^2 = 0.$$

Thus, the set $C_N$ is defined as a semi-algebraic subset of $(\mathbb{P}^3)^\binom{N}{2}$ by the following equations and inequalities:

1. The $\binom{N}{2}$ equations $\|v_{ij}\|^2 = \|v_{ji}\|^2$, see Section 2.3.2

2. For each of the $\binom{N}{3}$ triples $(i,j,k)$ the spherical triangle inequality, see Proposition 2.3.5

3. For each of the $6\binom{N}{4}$ ways to choose two triples of distinct indices $(i,j,k), (i,j,m)$ the spherical law of cosines compatibility, see Lemma 2.3.11
2.5 Proofs

Proof of Proposition 2.3.5. Fix representatives \((v_{ij}, v_{ji})\), \((v_{ik}, v_{ki})\) and \((v_{jk}, v_{kj})\).

Since the lengths \(\alpha_{ijk}, \beta_{ijk}, \gamma_{ijk}\) strictly satisfy the triangle inequalities, there is a non-degenerate spherical triangle with these edge lengths. Denote the vertex of this triangle opposite the edge of length \(\alpha_{ijk}\) by \(\Lambda_{jk}\), the vertex opposite the edge \(\beta_{ijk}\) by \(\Lambda_{ik}\) and the vertex opposite the edge \(\gamma_{ijk}\) by \(\Lambda_{ij}\). Since this triangle is non-degenerate, we know that \(\Lambda_{ij}, \Lambda_{ik}\) and \(\Lambda_{jk}\) are linearly independent. Thus we have embeddings \(\iota_i, \iota_j, \iota_k\), given by

\[
\begin{align*}
\iota_i &: P_i \hookrightarrow \mathbb{R}^3, \\
\iota_j &: P_j \hookrightarrow \mathbb{R}^3, \\
\iota_k &: P_k \hookrightarrow \mathbb{R}^3, \\
v_{ij} &\mapsto \Lambda_{ij}, \\
v_{ji} &\mapsto \Lambda_{ij}, \\
v_{ki} &\mapsto \Lambda_{ik}, \\
v_{kj} &\mapsto \Lambda_{kj}.
\end{align*}
\]

Observe that these embeddings are isometric by construction, and so \(F_i = (\iota_i(x), \iota_i(y)), F_j = (\iota_j(x), \iota_j(y))\) and \(F_k = (\iota_k(x), \iota_k(y))\) are frames. Since \(\Lambda_{ij}, \Lambda_{ik}, \Lambda_{jk}\) are vertices of a non-degenerate spherical triangle, these three frames are in generic position. Moreover, by construction we have

\[
\begin{align*}
\iota_i(v_{ij}) &= \iota_j(v_{ji}), \\
\iota_i(v_{ik}) &= \iota_k(v_{ki}), \\
\iota_j(v_{jk}) &= \iota_k(v_{kj}),
\end{align*}
\]

and so \(F_i, F_j\) and \(F_k\) realize the required common line pairs.

Now, suppose \(G_i, G_j, G_k\) also realize the common line pairs \([v_{ij} : v_{ji}], [v_{ik} : v_{ki}]\) and \([v_{jk} : v_{kj}]\). Let \(\iota_i^G, \iota_j^G\) and \(\iota_k^G\) be the embeddings corresponding to these frames, and set \(\Lambda_{ij}^G = \iota_i^G(v_{ij}), \Lambda_{ik}^G = \iota_i^G(v_{ik})\) and \(\Lambda_{jk}^G = \iota_j^G(v_{jk})\). Since \((i, j, k)\) strictly satisfies the triangle inequalities, these three vectors are linearly independent and thus define
a spherical triangle with edge lengths \((\alpha_{ijk}, \beta_{ijk}, \gamma_{ijk})\). This triangle is congruent to the triangle with vertices \(\Lambda_{ij}, \Lambda_{ik}, \Lambda_{jk}\) constructed above, and so there exists an element in \(O(3)\) that maps \(\Lambda^G_{ij} \mapsto \Lambda_{ij}, \Lambda^G_{ik} \mapsto \Lambda_{ik}\) and \(\Lambda^G_{jk} \mapsto \Lambda_{jk}\), and thus maps \((G_i, G_j, G_k) \mapsto (F_i, F_j, F_k)\).

\[ \square \]

**Proof of Lemma 2.3.11.** Fix unit length representatives \((v_{ij}, v_{ji}), (v_{ik}, v_{ki}), (v_{im}, v_{mi})\) and \((v_{jm}, v_{mj})\). Let \(\iota^F_i, \iota^F_j, \text{ and } \iota^F_k\) be the embeddings corresponding to \(F_i, F_j, F_k\), and let \(\iota^G_i, \iota^G_j, \iota^G_m\) be the embeddings corresponding to \(G_i, G_j, G_m\). Write

\[
\Lambda^F_{ij} = \iota^F_i(v_{ij}), \quad \Lambda^F_{ik} = \iota^F_i(v_{ik}), \quad \Lambda^F_{jk} = \iota^F_j(v_{jk}),
\]
\[
\Lambda^G_{ij} = \iota^G_i(v_{ij}), \quad \Lambda^G_{ik} = \iota^G_i(v_{ik}), \quad \Lambda^G_{jk} = \iota^G_j(v_{jk}).
\]

We wish to show that the map \(A : \mathbb{R}^3 \to \mathbb{R}^3\) defined by

\[
\Lambda^F_{ij} \mapsto \Lambda^G_{ij}, \quad \Lambda^F_{ik} \mapsto \Lambda^G_{ik}, \quad \Lambda^F_{jk} \mapsto \Lambda^G_{jk},
\]

which sends \(F_i \mapsto G_i\) and \(F_j \mapsto G_j\), is an isometry in \(O(3)\). Since \((i, j, k)\) is realized by \(F_i, F_j, F_k\) it satisfies the spherical triangle inequality, and thus the vectors \(\Lambda^F_{ij}, \Lambda^F_{ik}, \Lambda^F_{jk}\) are linearly independent and the map \(A\) is uniquely determined. We have that

\[
\Lambda^F_{ij} \cdot \Lambda^F_{ij} = \Lambda^G_{ij} \cdot \Lambda^G_{ij}, \quad \Lambda^F_{ik} \cdot \Lambda^F_{ik} = \Lambda^G_{ik} \cdot \Lambda^G_{ik}, \quad \Lambda^F_{jk} \cdot \Lambda^F_{jk} = \Lambda^G_{jk} \cdot \Lambda^G_{jk},
\]
and further, that

\[ \Lambda_{ij}^F \cdot \Lambda_{ik}^F = \iota_i^F (v_{ij}) \cdot \iota_i^F (v_{ik}) = v_{ij} \cdot v_{ik} = \iota_i^G (v_{ij}) \cdot \iota_i^G (v_{ik}) = \Lambda_{ij}^G \cdot \Lambda_{ik}^G, \]

\[ \Lambda_{ij}^F \cdot \Lambda_{jk}^F = \iota_j^F (v_{ji}) \cdot \iota_j^F (v_{jk}) = v_{ji} \cdot v_{jk} = \iota_j^G (v_{ji}) \cdot \iota_j^G (v_{jk}) = \Lambda_{ij}^G \cdot \Lambda_{jk}^G. \]

It follows that we only need to show that \( \Lambda_{ik}^F \cdot \Lambda_{jk}^F = \Lambda_{ik}^G \cdot \Lambda_{jk}^G \) to conclude that \( A \) is an isometry.

We first discuss the relative orientation of the common line pairs. The product

\[ \det[v_{ij}, v_{ik}] \det[v_{ij}, v_{im}] \]

is positive if the shortest rotation from \( v_{ij} \) to \( v_{ik} \) in \( P_i \) is in the same direction as the shortest rotation from \( v_{ij} \) to \( v_{im} \). In this case, we say \( v_{ik} \) and \( v_{im} \) lie on the same side of \( v_{ij} \). This product is negative if the shortest rotation from \( v_{ij} \) to \( v_{ik} \) is in the opposite direction of the shortest rotation of \( v_{ij} \) to \( v_{im} \), and, in this case, we say \( v_{ik} \) and \( v_{im} \) lie on opposite sides of \( v_{ij} \). Similarly, the sign of the product \( \det[v_{ji}, v_{jk}] \det[v_{ji}, v_{jm}] \) determines if \( v_{jk} \) and \( v_{jm} \) lie on the same, or opposite, sides of \( v_{ji} \) in \( P_j \).

Since we consider isometric embeddings of \( P_i \) and \( P_j \), we can make the same statements for the embedded versions of these vectors: \( \Lambda_{ik}^G \) and \( \Lambda_{im}^G = \iota_i^G (v_{im}) \) lie on the same side of \( \Lambda_{ij}^G \) in the plane \( \iota_i^G (P_i) \) if \( \det[v_{ij}, v_{ik}] \det[v_{ij}, v_{im}] \) is positive, and these vectors lie on opposite sides of \( \Lambda_{ij}^G \) if this product is negative. We can similarly say whether the vectors \( \Lambda_{jk}^G \) and \( \Lambda_{jm}^G = \iota_j^G (v_{jm}) \) lie on the same or opposite sides of \( \Lambda_{ij}^G \) in the plane \( \iota_j^G (P_j) \).

Next, consider the spherical triangle \( T \), with vertices \( \Lambda_{ij}^G, \Lambda_{ik}^G, \) and \( \Lambda_{jk}^G \), and the triangle \( T' \), with vertices \( \Lambda_{ij}^G, \Lambda_{im}^G, \Lambda_{jm}^G \). The triangles \( T \) and \( T' \) share the vertex \( \Lambda_{ij}^G \), and we write \( Z \) for the angle of \( T \) at this vertex and \( Z' \) for the angle of \( T' \) at this vertex.

Suppose first that \( \Lambda_{ik}^G \) and \( \Lambda_{im}^G \) both lie on the same side of \( \Lambda_{ij}^G \) in \( \iota_i^G (P_i) \), and \( \Lambda_{jk}^G \) and \( \Lambda_{jm}^G \) both lie on the same side of \( \Lambda_{ij}^G \) in \( \iota_j^G (P_j) \). In this case the triangles
T and T' sit inside each other, so Z and Z' are the same (cf. Figure 2.10a). On the other hand, if $\Lambda^G_{ik}$ and $\Lambda^G_{im}$ lie on opposite sides of $\Lambda^G_{ij}$, and $\Lambda^G_{jk}$, $\Lambda^G_{jm}$ also lie on opposite sides of $\Lambda^G_{ij}$, then the triangle T' lies opposite of T across $\Lambda^G_{ij}$, so the vertical angles Z and Z' are equal. These two cases occur if and only if the quantity

$$\sigma = \text{sign} (\det[v_{ij}, v_{ik}] \det[v_{ij}, v_{im}] \det[v_{ji}, v_{jk}] \det[v_{ji}, v_{jm}]),$$

is +1. Similarly, $\sigma = -1$ if and only if one of the pairs $\Lambda^G_{ik}$, $\Lambda^G_{im}$ or $\Lambda^G_{jk}$, $\Lambda^G_{jm}$ lies on the same side of $\Lambda^G_{ij}$, while the other pair lies on opposite sides of $\Lambda^G_{ij}$. In this case the triangles T and T' sit side by side, so the angles Z and Z' are supplementary (cf. Figure 2.10b).

It follows that $\cos Z = \sigma \cos Z'$, and so applying the spherical law of cosines in T yields

$$\frac{\Lambda^G_{ik} \cdot \Lambda^G_{jk} - (v_{ij} \cdot v_{ik})(v_{ji} \cdot v_{jk})}{| \det[v_{ij}, v_{ik}] \det[v_{ji}, v_{jk}] |} = \sigma \cos Z'.$$

On the other hand, the law of cosines in T' gives

$$\cos Z' = \frac{v_{mi} \cdot v_{mj} - (v_{ij} \cdot v_{im})(v_{ji} \cdot v_{jm})}{| \det[v_{ij}, v_{im}] \det[v_{ji}, v_{jm}] |},$$

and finally, since $L_{ijk,ijm} = 0$ we have

$$\sigma \cos Z' = \frac{v_{ki} \cdot v_{kj} - (v_{ij} \cdot v_{ik})(v_{ji} \cdot v_{jk})}{| \det[v_{ij}, v_{ik}] \det[v_{ji}, v_{jk}] |}.$$

Thus we have that $\Lambda^G_{ik} \cdot \Lambda^G_{jk} = v_{ki} \cdot v_{ki} = \Lambda^F_{ik} \cdot \Lambda^F_{jk}$, and so $A$ is an isometry, as desired.

\[\square\]

**Proof of Theorem 2.3.12.** By Proposition 2.3.3 we first obtain realizing frames $F_1$, $F_2$, $F_3$ for the triple (1, 2, 3). For all remaining indices $i$, we construct realizing
frames $G_1, G_2, G_i$ from the triple $(1, 2, i)$. By Lemma 2.3.11 there exists a unique map $A_i \in O(3)$ that maps $F_1 \mapsto G_1$ and $F_2 \mapsto G_2$. If $\det A_i = -1$ we can replace the realizing frames $G_1, G_2, G_i$ by $L(G_1), L(G_2), L(G_i)$, where $L$ is an arbitrary isometry in $O(3)$ with $\det L = -1$, and replace $A_i$ by $L \circ A_i$. It follows that we can assume $A_i$ has $\det = +1$. We set $F_i = A_i^{-1}G_i$.

Now we need to check that the $F_i$ are realizing frames. We will write $\iota_i^F$, $\iota_j^F$ and $\iota_k^F$ for the embeddings determined by $F_i, F_j, F_k$, and similarly for other sets of reconstructed frames. Thus, we need to verify that $\iota_i^F(v_{ij}) = \iota_j^F(v_{ji})$ for all pairs $i, j$.

To this end, suppose that $F_1 = A_i^{-1}G_i$ was reconstructed from $G_1, G_2, G_i$ and $F_2 = A_j^{-1}D_j$ was reconstructed from $D_1, D_2, D_j$. The triple $(1, i, j)$ also strictly satisfies the triangle inequality, so we have generic realizing frames $H_1, H_i, H_j$. By Lemma 2.3.11 we have isometries $B_i : (G_1, G_i) \mapsto (H_1, H_i)$ and $B_j : (D_1, D_j) \mapsto (H_1, H_j)$.

These maps and frames fit into the following diagram:

$$
\begin{array}{ccc}
(G_1, G_2, G_i) & \xrightarrow{A_i} & (F_1, F_2, F_3) \\
& \downarrow{B_i} & \downarrow{A_j} \\
(H_1, H_i, H_j) & \xrightarrow{B_j} & (D_1, D_2, D_j)
\end{array}
$$

First, note that $\det B_i = \pm 1$ and $\det B_j = \pm 1$, and in fact, we claim that $\det B_i = \det B_j$. To see why, write $\Lambda_{12}^G = \iota_1^G(v_{12}) = \iota_2^G(v_{21})$, and similarly for all other common line pairs. Then, we have

$$
\begin{align*}
(\det B_i) \text{ sign}(\det [\Lambda_{12}^G, \Lambda_{11}^G, \Lambda_{21}^G]) &= \text{ sign}(\det [\Lambda_{12}^H, \Lambda_{11}^H, \Lambda_{21}^H]), \\
(\det B_j) \text{ sign}(\det [\Lambda_{12}^D, \Lambda_{11}^D, \Lambda_{21}^D]) &= \text{ sign}(\det [\Lambda_{12}^H, \Lambda_{11}^H, \Lambda_{21}^H]).
\end{align*}
$$
Further, if \( \sigma = \text{sign} (\det [v_{12}, v_{11}] \det [v_{12}, v_{1j}] \det [v_{21}, v_{2i}] \det [v_{21}, v_{2j}]) \), we have

\[
\text{sign} (\det [\Lambda_{12}^H, \Lambda_{1i}^H, \Lambda_{2i}^H]) = \sigma \text{sign} (\det [\Lambda_{12}^H, \Lambda_{1j}^H, \Lambda_{2j}^H]),
\]

\[
\text{sign} (\det [\Lambda_{12}^G, \Lambda_{1i}^G, \Lambda_{2i}^G]) = \sigma \text{sign} (\det [\Lambda_{12}^G, \Lambda_{1j}^G, \Lambda_{2j}^G]),
\]

\[
\text{sign} (\det [\Lambda_{12}^D, \Lambda_{1i}^D, \Lambda_{2i}^D]) = \sigma \text{sign} (\det [\Lambda_{12}^D, \Lambda_{1j}^D, \Lambda_{2j}^D]).
\]

On the other hand, \( \det (A_i \circ A_j^{-1}) = 1 \), so

\[
\text{sign} (\det [\Lambda_{12}^G, \Lambda_{1i}^G, \Lambda_{2i}^G]) = \sigma \text{sign} (\det [\Lambda_{12}^D, \Lambda_{1j}^D, \Lambda_{2j}^D]),
\]

and thus \( \det B_i = \det B_j \). Note that the diagram above commutes, since both the top path and bottom path are morphisms in \( O(3) \) of the same determinant that send \( F_1 \mapsto H_1 \). Then, since \( H_1, H_i, H_j \) realize the common line pair \( (v_{ij}, v_{ji}) \), we have

\[
\iota_i^F (v_{ij}) = A_i^{-1} \iota_i^G (v_{ij}) = (A_i^{-1} \circ B_i^{-1}) \iota_i^H (v_{ij})
\]

\[
= (A_i^{-1} \circ B_i^{-1}) \iota_j^H (v_{ji})
\]

\[
= (A_j^{-1} \circ B_j^{-1}) \iota_j^H (v_{ji}) = A_j^{-1} \iota_j^D (v_{ji}) = \iota_j^F (v_{ji})
\]

and thus \( F_1, \ldots, F_N \) realize the common lines data \( ([v_{ij} : v_{ji}]) \).

Finally, suppose that \( F'_1, \ldots, F'_N \) is another collection of realizing frames for \( ([v_{ij} : v_{ji}]) \). Fix a triple \( (i, j, k) \), and observe that since both \( F_i, F_j, F_k \) and \( F'_i, F'_j, F'_k \) are realizing frame for \( (i, j, k) \) by Proposition 2.3.5 there is an isometry \( R_{ijk} \) that sends \( (F'_i, F'_j, F'_k) \mapsto (F_i, F_j, F_k) \). Note that for any \( i, j, k, m \), the two isometries \( R_{ijk} \) and \( R_{ijm} \) are equal since they agree on \( F'_i \) and \( F'_j \). This implies that \( R_{ijk} (F'_m) = F_m \) for all \( m \), and thus there is a single isometry \( (F'_1, \ldots, F'_N) \mapsto (F_1, \ldots, F_N) \).

\[\Box\]

**Proof of Theorem 2.3.15** First, observe that the minors corresponding to a common
line pair $[v_{ij} : v_{ji}]$ are non-zero for points in $\rho(\mathcal{G})$, since otherwise $F_i$ and $F_j$ would define the same plane. It follows that the rational projection $\text{Gr}(3,2N) \rightarrow (\mathbb{P}^3)^{N \choose 2}$ is defined everywhere on $\rho(\mathcal{G})$.

By definition any valid common lines data $(v_{ij} : v_{ji}) \in C_N$ has some realizing frames $F_1, \ldots, F_N$, and so is the image of $\pi(\rho(F_*))$ and thus $\pi(\rho(\mathcal{G})) = C_N$. It only remains to verify that this projection is injective. This follows from Theorem 2.3.12. If $\pi(\rho(F_*)) = \pi(\rho(G_*))$, then we know that the realizing frames $F_*$ and $G_*$ are related by an isometry in $O(3)$. But then the rows of the matrices $F_*$ and $G_*$ define the same linear subspace, and so $\rho(F_*) = \rho(G_*)$.

\[ \square \]

**Proof of Corollary 2.3.16** We will compute dimensions with respect to a dense subset of $\mathcal{G}$ and a dense subset of $\rho(\mathcal{G}) \times O(3)$. Let $V \subset \mathcal{G}$ be the complement of the semi-algebraically homeomorphic to an open subset of $\text{SO}(3)^N$ we have $\dim V = \dim \mathcal{G} = 3N$, and thus $\dim \rho(V) = \dim \rho(\mathcal{G}) = 3N - 3$. By Theorem 2.3.15 we have a semi-algebraic bijection between $\rho(\mathcal{G})$ and $C_N$, so we conclude that $\dim C_N = 3N - 3$.

\[ \square \]
A symmetry of a biological molecule is a proper rotation of three-dimensional space that leaves the molecule unchanged. For example, Figure 3.1a is a 3D structure of Env, the HIV-1 envelope glycoprotein, discovered by Lyumkis et al [12]. This protein is unchanged if we rotate it by $2\pi/3$ radians about the axis $A$—see Figure 3.1b for a view of Env along $A$. We say Env has a three-fold symmetry about the symmetry axis $A$.

The collection of all symmetries of a molecule is called the point group of the molecule. For Env, if we apply the above rotation again, now rotating the molecule by $4\pi/3$ about $A$, Env is once more unchanged. Thus, rotation by $4\pi/3$ about $A$ is another symmetry of Env. This protein does not have any other symmetries, so we
conclude that the point group of Env, denoted $\Gamma$, is given by

$$\Gamma = \{I, R, R^2\}, \quad (3.1.1)$$

where $I$ is the identity rotation\footnote{Recall that every group must contain an identity element. We think of the identity $I$ as a rotation by 0 radians.} and $R$ is rotation by $2\pi/3$ about the axis $A$.

The point group of a molecule contributes to its biological function, and discovering this group is of fundamental interest when studying a symmetric molecule. Furthermore, knowing the point group of a molecule is often an important ingredient in \textit{three-dimensional reconstruction}, which is the process of generating a three-dimensional model of a molecule from an experiment. For example, commonly used three-dimensional reconstruction algorithms used in cryo-electron microscopy (cryo-EM) require knowing the point group of a molecule to perform reconstruction\cite{fehringer2016three, becker2008three}.

### 3.1.1 Detecting 3D molecular symmetries from 2D data

Motivated by the discussion above, we study the problem of determining a molecular point group from suitable two-dimensional projections of the molecule, for example...
from cryo-EM images. This project is joint with Yoel Shkolnisky and Shamgar Gurevich.

**Problem 3.1.2** (Point Group Detection). *Given noisy two-dimensional cryo-EM images of a molecule, determine the molecule’s three-dimensional point group.*

**Detecting cyclic point groups**

We will first discuss how one can solve Problem 3.1.2 for simple types of point groups. We saw above that the point group \( \Gamma \) of Env, see Eq. 3.1.1, consists of rotations by \( \frac{2\pi}{3} \) and \( \frac{4\pi}{3} \) about the axis \( A \), Figure 3.1b. This point group is the cyclic group with 3 elements, denoted \( C_3 \). A cyclic point group consists of \( n \) successive rotations by \( \frac{2\pi}{n} \) about a single axis \( A \), and is denoted \( C_n \). Figure 3.2 shows several examples of objects with cyclic point groups.

![Figure 3.2: Objects with cyclic point groups. Images by Emmanuel Levy.](image)

**Seeing cyclic 3D symmetries in 2D projections**

We now outline one way to recover a cyclic point group from 2D projections. Suppose we are studying a molecule with point group \( \Gamma = C_n \), consisting of rotations about some axis \( A \), and we obtain a cryo-EM image by projecting along Env’s symmetry axis—Figure 3.3a. This image—Figure 3.3b—also has symmetries given by rotations of \( \frac{2\pi}{3} \) and \( \frac{4\pi}{3} \). We say that this image has *rotational symmetry order* 3.
In other words, the 2D symmetries of this cryo-EM image (obtained by projecting along $A$) witness the three-fold 3D symmetries about the axis $A$ in the point group of the molecule.

**Detecting non-cyclic point groups**

Cyclic point groups consist of rotations about a single axis $A$, while more complicated point groups consist of rotations around more than a single axis.

For example, Figure 3.4a is a view of the protein GroEL, due to Ludtke et al [11]. GroEL has a 7-fold symmetry: the molecule is unchanged if we rotate it by $2\pi/7$ about the axis $A$, see Figure 3.4b. However, GroEL has additional symmetries given by rotating by $2\pi/2$ about the axis $B$, see Figure 3.4c.

Point groups that consist of an $n$-fold symmetry about an axis $A$, together with 2-fold symmetries about axes $B$ perpendicular to $A$, are called dihedral groups and are denoted $D_n$. Thus, the point group of GroEL is $D_7$. Figure 3.5 gives examples of several objects with dihedral symmetry. Each of these objects has $n$-fold symmetry about the axis $A$, and 2-fold symmetry about the axis $B$.
We saw above that the protein Env produces images with rotational symmetry order 3 when we project the molecule along the symmetry axis $A$. Images of GroEL will exhibit two different rotational symmetry orders: we obtain images with rotational symmetry order 7 by projecting along $A$, Figure 3.4b, as well as images with rotational symmetry order 2 by projecting along $B$, Figure 3.4c.

### 3.1.2 Flowchart Algorithm

We now describe how the observations from the previous section can be used to describe a simple "Flowchart" algorithm that will detect the point groups of Env and GroEL. Env has images with rotational symmetry order 3, Figure 3.1b, whereas images of GroEL will exhibit rotational symmetry orders 7, Figure 3.4b, and rotational...
symmetry order 2, Figure 3.4c.

Figure 3.6: Flowchart algorithm for determining the point groups of Env and GroEL

This suggests a flowchart algorithm, Figure 3.6 to detect the point groups $\Gamma = C_3$ of Env and $\Gamma = D_7$ of GroEL. Our input consists of many noisy cryo-EM images of one of these molecules. We first compute the rotational symmetry order of each image, assigning a rotational symmetry order of 1 to asymmetric images.

Input images obtained from near symmetry axes of the molecule will have rotational symmetry orders greater than 1. After we compute all rotational symmetry orders, we examine which orders appear. If the only order that appears is 3, we conclude our molecule’s point group is $C_3$. On the other hand, if the rotational symmetry orders 2 and 7 appear, we conclude that the molecule’s point group is the dihedral group $D_7$.

This algorithm works because molecules with $C_3$ and $D_7$ symmetry are distinguished by the rotational symmetry orders appearing in images. In fact, any cyclic group $C_n$ and $D_n$ can be distinguished in this way: we expect that molecules with cyclic point groups will only produce images with rotational symmetry order $n$, whereas molecules with dihedral groups will produce images with rotational symmetry orders 2 and $n$. Thus we can extend this version of the flowchart algorithm to detect all cyclic groups $C_n$, and all dihedral groups $D_n$ for $n > 3$.

\footnote{When $n = 2$ both $C_2$ and $D_2$ produce only images with rotational symmetry order 2, so these groups must be distinguished in another way. We describe this ambiguity in more detail later.}
3.1.3 Our contribution and prior work

In the remainder of this chapter we will see that all possible molecular point groups, not just cyclic and dihedral groups, can be distinguished by the rotational symmetry orders in images of the molecule. Thus, we will be able to extend the flowchart, Figure 3.6, to cover all point groups.

We believe that the essential idea of deducing the point group of a molecule by looking at the rotational symmetry orders of images is part of the folklore knowledge in the electron microscopy community. For example, Crowther defines the rotational power spectrum and uses it to compute the rotational symmetry order of individual images. Danziger et al., Figure 5 in [2], analyze the 7-fold symmetry of a GroEL mutant by examining an image with rotational symmetry order 7. Kocsis et al., [9], describe a statistical approach for analyzing rotational symmetry orders of large collections of images, and discuss inferring the presence of icosahedral symmetry from multiple different top views. However, we were surprised to have not found any complete reference explaining how the rotational symmetry orders of images determine the 3D point group of the molecule.

Thus, this work provides two new contributions: first, we present a complete, self-contained explanation of this algorithm; second, we describe a reference implementation of the algorithm, and present numerical results validating the algorithm on simulated data.

3.2 Flowchart algorithm for detecting point groups

In this section we give a complete description of the flowchart algorithm to detect molecular point groups.
Figure 3.7: Objects exhibiting each possible point group

3.2.1 Complete list of point groups

Before describing the flowchart algorithm, we first recall the different types of point groups that can appear and that our algorithm must detect. A symmetry of an object in space is a proper rotation that leaves the object unchanged. The collection of all symmetries of an object is called the point group of the object. We only consider point groups consisting of finitely many proper rotations. Figure 3.7 shows example objects for each type of point group.
Cyclic Groups

The simplest type of point group is the cyclic group with \( n \) elements, denoted \( C_n \), see Figure 3.7a. This point group consists of successive rotations by \( 2\pi/n \) about a single axis \( A \). We say that the point group \( C_n \) contains symmetries of order \( n \).

Dihedral Groups

Next, the dihedral point groups, denoted \( D_n \), contain \( n \) successive rotations by \( 2\pi/n \) about an axis \( A \), together with rotations by \( 2\pi/2 \) about axes \( B \) perpendicular to \( A \), see Figure 3.7b. A dihedral point group contains symmetries of order \( n \) (about \( A \)) and symmetries of order 2 (about \( B \)).

Symmetry groups of platonic solids

The tetrahedral point group, denoted \( T \), is the collection of all symmetries of a tetrahedron centered at the origin in space. This point group contains rotations by \( 2\pi/3 \) about axes \( A \) through the center of each triangular face and rotations by \( 2\pi/2 \) about axes \( B \) joining midpoints of opposite edges. Thus, the symmetry orders appearing in \( T \) are 3 and 2.

The octahedral point group, denoted \( O \), is the collection of all symmetries of a cube centered at the origin in space. This group contains rotations by \( 2\pi/4 \) about axes \( A \) through the center of each square face, rotations by \( 2\pi/3 \) about axes \( B \) along the diagonals of the cube, and rotations by \( 2\pi/2 \) about axes \( C \) joining the midpoints of opposite edges. The symmetry orders appearing in \( O \) are 4, 3 and 2.

Finally, the icosahedral point group, denoted \( I \), is the collection of all symmetries of an icosahedron. This group contains rotations by \( 2\pi/5 \) about axes \( A \) through each pair of opposing vertices, rotations by \( 2\pi/3 \) about axes \( B \) through the centers.

\( ^3 \)The cube and octahedron have the same symmetry group; we find it easier to visualize this group using a cube, but the established notation is \( O \).
of opposing triangles, and rotations by $2\pi/2$ about axes $C$ through midpoints of opposite edges. The symmetry orders in the point group $I$ are 5, 3 and 2.

### 3.2.2 Distinguishing point groups

Table 3.1 summarizes the different kinds of symmetries in each point group described above. Examining this table we can see that each point group is distinguished by the orders of symmetry that appear in the group. For instance, $D_7$ is the only group in Table 3.1 with 2-fold and 7-fold symmetry.

<table>
<thead>
<tr>
<th>Group</th>
<th>Symmetry orders</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_n$</td>
<td>$n$-fold about $A$</td>
</tr>
<tr>
<td>$D_n$</td>
<td>$n$-fold about $A$; 2-fold about $B$</td>
</tr>
<tr>
<td>$T$</td>
<td>3-fold about $A$; 2-fold about $B$</td>
</tr>
<tr>
<td>$O$</td>
<td>4-fold about $A$; 3-fold about $B$; 2-fold about $C$</td>
</tr>
<tr>
<td>$I$</td>
<td>5-fold about $A$; 3-fold about $B$; 2-fold about $C$</td>
</tr>
</tbody>
</table>

An important fact is that this list of point groups is complete. A theorem due to Felix Klein tells us that,

**Fact 3.2.1 (Klein’s Theorem (1884)).** *The only possible point groups in three-dimensional space are*[^1]  

1. *the cyclic groups* $C_n$,
2. *the dihedral groups* $D_n$,
3. *the symmetries of the tetrahedron, octahedron and icosahedron.*

Thus, we can use Table 3.1 to distinguish between all possible point groups.

[^1]: Recall that we only consider point groups consisting of finitely many proper rotations
3.2.3 Detecting point groups

The key observation for our algorithm is that we expect a cryo-EM projection of a molecule along an n-fold symmetry axis to produce an image with rotational symmetry order n. Since the point groups in Table 3.1 are distinguished by the orders of symmetry, it follows that molecular point groups can be distinguished by seeing which rotational symmetry orders appear in images of the molecule.

For example, in images of the protein Env, we will detect images with rotational symmetry order 3, Figure 3.3b, obtained by projecting along the axis A, see Figure 3.1a. Consulting Table 3.1, we see that $C_3$ is the only group with exclusively three-fold symmetries, so we conclude that the point group of Env is $C_3$.

On the other hand, in a dataset of GroEL cryo-EM images Figure 3.8a, we will detect images with rotational symmetry order 7, see Figure 3.8b and rotational symmetry order 2, see Figure 3.8c. The only group in Table 3.1 with these symmetry orders is the dihedral group $D_7$, so we conclude that this is GroEL’s point group.

3.2.4 Flowchart Algorithm Summary

Algorithm 1 summarizes the flowchart algorithm we have described.
Algorithm 1 Complete Flowchart Algorithm

1: **INPUT:** $I_1, \ldots, I_N$ \> Cryo-EM images of a molecule
2: for $i \leftarrow 1, \ldots, N$ do
3: \quad rot_sym_ords$_i$ $\leftarrow$ rotational symmetry order of $I_i$
4: end for
5: 
6: if only $n$ in rot_sym_ords, for $n > 5$ then
7: \quad **OUTPUT:** $C_n$
8: else if $2, n$ in rot_sym_ords, for $n > 5$ then
9: \quad **OUTPUT:** $D_n$
10: else if $2, 4$ in rot_sym_ords then
11: \quad if $3$ in rot_sym_ords then
12: \quad \quad **OUTPUT:** $O$
13: \quad else
14: \quad \quad **OUTPUT:** $D_4$
15: \quad end if
16: else if $2, 3$ in rot_sym_ords then
17: \quad if $5$ in rot_sym_ords then
18: \quad \quad **OUTPUT:** $I$
19: \quad else
20: \quad \quad **OUTPUT:** $D_3/T$
21: \quad end if
22: else if $2$ in rot_sym_ords then
23: \quad **OUTPUT:** $C_2/D_2$
24: else
25: \quad **OUTPUT:** 1 \> No symmetries, return the trivial symmetry group
26: end if
$C_2/D_2$ and $T/D_3$ ambiguities

Lines 20 and 23 of Algorithm 1 each output two possible groups. We can see this ambiguity already in Table 3.1: molecules with point groups $C_2$ and $D_2$ both produce images only with rotational symmetry order 2, while molecules with point groups $D_3$ and $T$ both produce images with rotational symmetry orders 2 and 3. Since our algorithm deduces the point group from these symmetry orders, it will not be able to tell these groups apart. Thus, to distinguish these point groups we must use additional information.

For example, to resolve between $C_2$ and $D_2$ we could compare all images with rotational symmetry order 2. In the case of $C_2$ there should be only a single view of the molecule among these images, while we expect a molecule with $D_2$ symmetry to have two distinct views with rotational symmetry order 2.

If we only observe images with rotational symmetry order 2 and 3, we need to distinguish between $D_3$ and $T$. We suggest to estimate the relative projection directions of the 2-fold and 3-fold images. In the case of the dihedral group $D_3$, these projection directions are perpendicular, see Figure 3.5, while for the point group $T$ the angle between these axes is $\cos^{-1}(-1/\sqrt{3}) \approx 125.3$ degrees.

False symmetries

As we discussed above, we expect that a cryo-EM image obtained along an $n$-fold symmetry axis will have rotational symmetry order $n$. This is not, however, a guarantee: in general, an image obtained along an $n$-fold axis will have rotational symmetry order $m$, where $n$ divides $m$. For example, a projection of a cube along one of its diagonals, a 3-fold axis, produces a hexagon, which has rotational symmetry order 6. This is due to the fact that cryo-EM images lose information about displacements along the projection direction. In this example, since there is no group in Table 3.1
with symmetry orders 2, 4 and 6, we can still correctly deduce the octahedral point group \( O \). However, it is possible to imagine 3D shapes with symmetries that cannot be detected from cryo-EM projections. For example, suppose we first place three spheres on the unit circle in the \( xy \) plane at 0, \( 2\pi/3 \) and \( 4\pi/3 \), and then shift one of the spheres to \( z = 1 \). This configuration has no 3D symmetries, but a cryo-EM image obtained by projecting along the \( z \) axis will produce an image with rotational symmetry order 3. So far we have not observed such adversarial molecules in our work.

**Other types of projections**

Finally, we note that the flowchart algorithm is not specific to cryo-EM. We can use this algorithm to detect the symmetries of a molecule from any type of projection that produces an image with rotational symmetry order \( n \) when we project along an \( n \)-fold 3D symmetry axis.

### 3.3 Implementation and numerical analysis on noisy data

In this section we describe our reference implementation of the above algorithm, which we call \textsc{FlowSym}. We validate the performance of our implementation on a benchmark set of simulated cryo-EM images of several molecules with cyclic and dihedral point groups under varying amounts of noise.

Our implementation is written in Python, using the NumPy and SciPy numerical and scientific computing packages. These packages contain a large number of general purpose utilities that allowed us to rapidly implement the algorithm and quickly experiment with different approaches for handling noisy images.
3.3.1 Validation Method

We validate FLOWSYM by using it to compute the point groups of a benchmark set of molecules with known point groups—see Table 3.2. Density maps of these molecules were obtained from the EM Data Bank (EMDataBank.org) [10].

Table 3.2: FLOWSYM test benchmark

<table>
<thead>
<tr>
<th>Molecule</th>
<th>EMDB ID</th>
<th>Source</th>
<th>Symmetry</th>
</tr>
</thead>
<tbody>
<tr>
<td>fungal fatty acid synthase</td>
<td>1338</td>
<td>[8]</td>
<td>$D_3$</td>
</tr>
<tr>
<td>Env</td>
<td>5779</td>
<td>[12]</td>
<td>$C_3$</td>
</tr>
<tr>
<td>clathrin-auxilin cage</td>
<td>2410</td>
<td>[22]</td>
<td>$D_6$</td>
</tr>
<tr>
<td>anthrax toxin PA63</td>
<td>5215</td>
<td>[13]</td>
<td>$C_2$</td>
</tr>
<tr>
<td>yeast fatty acid synthase</td>
<td>1623</td>
<td>[4]</td>
<td>$D_3$</td>
</tr>
</tbody>
</table>

We used the ASPIRE [15] software package to generate a large number of cryo-EM images of each molecule from uniformly distributed projection directions. To simulate noise, we add a noise image $N$ to the clean simulated projection $I$. Each pixel of the noise image was independently sampled from a mean-zero Gaussian distribution. The variance of this noise distribution was set according to the desired signal to noise ratio,

$$\text{SNR} = \frac{\text{var}(I)}{\text{var}(N)}.$$  

3.3.2 Rotational Symmetry Order

The first step in FLOWSYM is to compute the rotational symmetry order of each input image. Recall that the rotational symmetry order of an image is the number of times the image matches with itself as we rotate it about its center. For example, Figure 3.3b has rotational symmetry order 3.
Alignment of images

In order to correctly compute rotational symmetry orders, it is necessary that the input images be aligned so that the center of each image corresponds to a single point of the 3D molecule, for instance, see the discussion in [1], Section 2b. Performing this image alignment is a common pre-processing step implemented in many EM software packages, for example [18] and [14]. Thus, to keep our presentation simple, we assume that our input images have already been aligned.

Rotational auto-correlation

A natural choice to measure rotational symmetry order is to compute the rotational auto-correlation of the image. The rotational auto-correlation of an image \( I \) is the cross-correlation of \( I \) with the rotation of \( I \) by some angle \( \theta \), given by

\[
RAC(\theta) = \langle I, \text{rot}_\theta(I) \rangle,
\]

where the \( \text{rot}_\theta(I) \) is \( I \) rotated by \( \theta \) degrees, and \( \langle \ , \ \rangle \) is the standard dot product, interpreting the images as flattened vectors.

We normalize the images to have mean 0 and unit length prior to computing \( RAC(\theta) \), which makes our definition of rotational auto-correlation coincide with the usual Pearson correlation coefficient of the two images.

As we rotate the image, this auto-correlation will have maximums each time the image matches up with itself, see Figure 3.9b. We count the number of these peaks to compute the rotational symmetry order of the image.

Rotating an image

When we rotate a discrete image, the pixels of the rotated image will not line up with the pixels of the original image. Thus, producing a rotated image requires averaging
nearby pixels to assign a value to each rotated pixel. This is well studied problem, and the \texttt{scipy.ndimage.interpolation} package implements several different interpolation algorithms. For our validation we chose to use bilinear interpolation.

\textbf{Peak counting}

Once we compute the rotational auto-correlation of an image, we need to count the number of peaks in this signal to obtain the image’s rotational symmetry order. From clean images, Figure 3.9a, we obtain an auto-correlation signal, Figure 3.9b, that has sharp peaks easily distinguished by, for example, finding the largest Fourier coefficient of the signal. When processing images with noise, Figure 3.10a, the auto-correlation signal is degraded and counting the peaks is more difficult.

We experimented with several different approaches and found that a simple template matching approach worked best on our benchmark set. For each rotational symmetry order $k$ we would like to detect, we generate a template signal $T_k$. We then match the noisy auto-correlation against each template by cross-correlation.
We declare that an image has rotational symmetry order $k$ if its rotational auto-correlation matches the template $T_k$ best.

The template $T_k$ is a sequence of Gaussian peaks centered at $\{0, 2\pi/k, 2 \cdot 2\pi/k, \ldots, (k - 1) \cdot 2\pi/k\}$. Ideally we would set the width of our template peaks so that they closely matched the molecule’s true rotational auto-correlation functions. Unfortunately, we do not know what our molecule’s rotational auto-correlation is supposed to look like. In fact, the shape of the molecule’s true rotational auto-correlation depends on the shape of the molecule itself, so choosing a single collection of templates $T_k$ that will work for many different molecules is problematic.

Nevertheless, we found that setting the width of the template peaks in $T_k$ to the minimum of $360/(2k)$ degrees and $360 \cdot 0.05 = 18$ degrees produced templates good enough to correctly identify the symmetry groups in our benchmark.

---

5We say the width of a Gaussian peak is the area around the peak where the Gaussian is above 0.1% of its max
Excluding small angles

Under our noise model, for $\theta = 0$ the rotational auto-correlation of an image $I$ is given by

$$RAC(0) = \langle I + N, I + N \rangle = \langle I, I \rangle + 2\langle I, N \rangle + \langle N, N \rangle.$$

The $\langle N, N \rangle$ term in this expression is the variance of the noise. Thus, at small values of $\theta$ and at low SNR, this noise term will be substantially larger than the true rotational symmetry order of an image, we clip the rotational auto-correlation near $\theta = 0$ and $\theta = 360$. If we expect a maximum rotational symmetry order of $k$, then the first place a peak might occur is $360/k$. Thus, for $\theta < k/360$ and $\theta > (k - 1)360/k$, we set $RAC(\theta)$ to the maximum of $RAC(\theta)$ on $360/k \leq \theta \leq (k - 1)360/k$.

Finding significant images

In order to correctly identify a molecular point group, we assume that the input to FlowSym contains images obtained from close to each symmetry axis of the molecule. Other than this, we do not assume anything else about the input.

In particular, most images processed by FlowSym will be from projection directions that do not correspond to any symmetry axis. In the presence of noise, it is difficult to distinguish such asymmetric images from true symmetric images that give us information about the molecule’s point group.

To reduce the number of erroneously detected rotational symmetry orders we accumulate from asymmetric images, we only process images we believe have non-trivial rotational symmetry order. FlowSym considers an image to have significant symmetry if the image’s rotational auto-correlation only matches a single template $T_k$ strongly. For our validation test, we define this to mean that the second-best
template match is less than 80% of the best template match.

**Table lookup with voting**

Once FlowSym computes the rotational symmetry order of each image, we lookup the corresponding point group in Table 3.1. In theory, we would expect an unambiguous list of rotational symmetry orders - for example, in images of GroEL we would expect every image to either be asymmetric, or have symmetry order 2 or 7.

In practice, due to noise, we will detect false symmetry orders. To resolve this we apply the flowchart to only the most commonly observed rotational symmetry orders.

### 3.3.3 Numerical results

Table 3.3 reports the validation of FlowSym against each molecule in our benchmark set. For each molecule we processed 10000 simulated cryo-EM images from uniformly distributed projection directions. Table 3.3 lists the number of images detected for each rotational symmetry order at an SNR of 1/4 and 1/8. In this benchmark, FlowSym searches for rotational symmetry orders up to 10.

Once we compute the rotational symmetry orders present, we deduce the molecule's point group by comparing the rotational symmetry orders with Table 3.1.

The result of applying the flowchart algorithm is reported in the last column. FlowSym correctly determined the point group of every molecule in our test set at an SNR of 1/4 and 1/8.
Table 3.3: FlowSym Benchmark Results

<table>
<thead>
<tr>
<th>Molecule</th>
<th>EMDB ID</th>
<th>True Sym.</th>
<th>SNR</th>
<th>Rotational Symmetry Orders</th>
<th>Deduced Sym.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>GroEL</td>
<td>5001</td>
<td>$D_7$</td>
<td>1/4</td>
<td>278</td>
<td>3523</td>
</tr>
<tr>
<td>anthrax toxin</td>
<td>5215</td>
<td>$C_2$</td>
<td>1/4</td>
<td>7066</td>
<td>258</td>
</tr>
<tr>
<td>Env</td>
<td>5779</td>
<td>$C_3$</td>
<td>1/4</td>
<td>7887</td>
<td>-</td>
</tr>
<tr>
<td>fungal</td>
<td>1338</td>
<td>$D_3$</td>
<td>1/4</td>
<td>39</td>
<td>8335</td>
</tr>
<tr>
<td>yeast</td>
<td>1623</td>
<td>$D_3$</td>
<td>1/4</td>
<td>151</td>
<td>5042</td>
</tr>
<tr>
<td>clathrin</td>
<td>2410</td>
<td>$D_6$</td>
<td>1/4</td>
<td>1077</td>
<td>-</td>
</tr>
</tbody>
</table>

### 3.4 Conclusion & Future work

We have described a Flowchart algorithm for detecting the point group of a biological molecule from suitable projections, such as cryo-EM images. Our algorithm uses 2D symmetries in images to witness the existence of 3D symmetries in the molecule, and Klein’s Theorem tells us that this information is enough to determine the entire point group.

We validate a reference implementation of the Flowchart algorithm against simulated data of a small benchmark set of molecules with cyclic and dihedral symmetries. Our reference implementation performs well on simulated data, correctly detecting the point groups from a benchmark set corrupted by noise.

Although these initial results are promising, testing the algorithm on real cryo-EM images is necessary to properly evaluate its performance.
Bibliography


