INTRODUCTION

Distance Geometry Problems:
+ There are various problems that come under this heading.
+ In general, we are given a set of distances between atoms and we are required to compute the coordinates of all atoms.
+ Computed coordinates are typically with respect to a frame of reference that has its origin at some pre-specified atom.

Notation:
+ The coordinates of the atoms are $\{x^{(i)}| i = 1, 2, \ldots, n\}$.
  * So, we are dealing with a molecule that has $n$ atoms.
  * We want to calculate these $x^{(i)}$, when given all or some subset of:
+ $d_{i,j} = \|x^{(i)} - x^{(j)}\| \quad 1 \leq i, j \leq n$. 
INTRODUCTION

Variants of the problem differ with respect to:

- **Number of distances given:**
  - The full set of “n choose 2” distances makes the problem fairly easy to solve.
  - In most practical applications we are only given a $O(n)$ sparse set of distances.

- **Accuracy of distances given:**
  - Distances may be considered as exact, or
  - The problem may specify upper and lower bounds on distances.
  - Distance may be given as probability distributions.

So, there are four types of problems:

- **P1:** A complete set of exact distances
- **P2:** A complete set of approximate distances
- **P3:** A sparse set of exact distances
- **P4:** A sparse set of approximate distances.
MOTIVATION (1)

✶ NMR

+ While most of the protein structures in the PDB have been computed using X-ray analysis, about 15% have been determined by NMR (Nuclear Magnetic Resonance).

+ Advantages:
  ✶ Unlike X-ray analysis, NMR does not require crystals – the proteins may be in solution.
  ✶ Consequently, we can have more confidence that their conformations are close to that present in the cytosolic environment.

MOTIVATION (2)

✶ NMR

+ Disadvantages:
  ✶ NMR experiments only report distances between atoms.
    ✶ The atoms must be close to one another (typically within 5 Å of each other).
      ✶ This means the number of distances is much less than \( n \) choose 2.
    ✶ These distances have some experimental error.
      ✶ This means we have reduced accuracy.
  ✶ NMR can only be used for shorter proteins.
    ✶ “Short proteins” means less than a few hundred amino acids.
Recall, we work in a 3D Euclidean vector space:

- The coordinates of the atoms are \( \{x^{(i)} | i = 1, 2, ..., n\} \).
  - So, we are dealing with a molecule that has \( n \) atoms.
  - We want to calculate these \( x^{(i)} \), when given all or some subset of the inter-atomic distances:
    \[
    d_{ij} = \| x^{(i)} - x^{(j)} \|, \quad 1 \leq i, j \leq n.
    \]

- An \( n \) by \( n \) matrix \( X \) is used to store the \( x^{(i)} \) vectors in a column by column fashion.
  - The \( n \) by \( n \) symmetric matrix \( D \) holds the squares of distances:
    \[
    \{D\}_{ij} = d_{ij}^2.
    \]
  - The Gram matrix \( G \) is defined by:
    \[
    \{G\}_{ij} = x^{(i)T}x^{(j)}.
    \]
  - We use \( \hat{d}_{ij} \) to represent an approximation of the distance \( d_{ij} \) and we set
    \[
    \{\hat{D}\}_{ij} = \hat{d}_{ij}^2.
    \]

**A USEFUL IDENTITY** (4)

- Given all the \( d_{ij}^2 \) values, we can compute a Gram matrix using the “double centering formula”:
  \[
  \frac{1}{2} \left[ \frac{1}{n} \left( \sum_{a=1}^{n} d_{aa}^2 + \sum_{\beta=1}^{n} d_{\beta\beta}^2 \right) - d_y^2 - \frac{1}{n^2} \sum_{a=1}^{n} \sum_{\beta=1}^{n} d_{a\beta}^2 \right] = x^{(i)T}x^{(j)} = G_{ij}.
  \]

- **Note:** For consistency and clarity we will always use \( i \) and \( j \) to index a matrix entry while \( \alpha \) and \( \beta \) are used for summation indices.
A USEFUL IDENTITY (2)

**Proving the Double Centering Formula:**

+ Start with: \[ d_{ij}^2 = \|x^{(i)} - x^{(j)}\|^2 \]
  \[ = (x^{(i)} - x^{(j)})^T (x^{(i)} - x^{(j)}) \]
  \[ = x^{(i)T}x^{(i)} - 2x^{(i)T}x^{(j)} + x^{(j)T}x^{(j)}. \]

+ We will further assume that the \( x^{(i)} \) vectors have their centroid at the origin. This means:
  \[ \sum_{a=1}^{n} x^{(a)} = 0. \]

**Note:**

\[ x^{(i)T} \sum_{a=1}^{n} x^{(a)} = 0 \Rightarrow \sum_{a=1}^{n} x^{(a)T} x^{(a)} = 0 \Rightarrow \sum_{a=1}^{n} x^{(a)T} x^{(j)} = 0. \]

A USEFUL IDENTITY (3)

**Proving the Double Centering Formula:**

+ Since \[ d_{ij}^2 = x^{(i)T}x^{(i)} - 2x^{(i)T}x^{(j)} + x^{(j)T}x^{(j)} \] we get:

\[ \sum_{a=1}^{n} d_{ai}^2 = \sum_{a=1}^{n} x^{(a)T}x^{(a)} - 2 \sum_{a=1}^{n} x^{(a)T}x^{(i)} + \sum_{a=1}^{n} x^{(a)T}x^{(j)} = \sum_{a=1}^{n} x^{(a)T}x^{(a)} + nx^{(i)T}x^{(i)}. \]

+ Similarly:

\[ \sum_{\beta=1}^{n} d_{j\beta}^2 = \sum_{\beta=1}^{n} x^{(\beta)T}x^{(\beta)} + nx^{(j)T}x^{(j)}. \]

+ Combining:

\[ \sum_{a=1}^{n} d_{ai}^2 + \sum_{\beta=1}^{n} d_{j\beta}^2 = 2 \sum_{a=1}^{n} x^{(a)T}x^{(a)} + nx^{(i)T}x^{(i)} + nx^{(j)T}x^{(j)}. \]
A USEFUL IDENTITY (4)

Take the last line of the previous slide and multiply through by \( n \):

\[
n \left( \sum_{a=1}^{n} d_{a}^{2} + \sum_{\beta=1}^{n} d_{a}^{2} \right) = 2n \sum_{a=1}^{n} x^{(a)T} x^{(a)} + n^2 \left( x^{(i)T} x^{(i)} + x^{(j)T} x^{(j)} \right).
\]

Also, from last slide:

\[
\sum_{\beta=1}^{n} d_{a}^{2} = \sum_{\beta=1}^{n} x^{(\beta)T} x^{(\beta)} + nx^{(i)T} x^{(i)}.
\]

So:

\[
\sum_{a=1}^{n} \sum_{\beta=1}^{n} d_{a\beta}^{2} = \sum_{a=1}^{n} \sum_{\beta=1}^{n} x^{(\beta)T} x^{(\beta)} + n \sum_{a=1}^{n} x^{(a)T} x^{(a)} = 2n \sum_{a=1}^{n} x^{(a)T} x^{(a)}.
\]

Subtract this from the first equation:

\[
\left( \sum_{a=1}^{n} d_{a}^{2} + \sum_{\beta=1}^{n} d_{a}^{2} \right) - \sum_{a=1}^{n} \sum_{\beta=1}^{n} d_{a\beta}^{2} = n^2 \left( x^{(i)T} x^{(i)} + x^{(j)T} x^{(j)} \right).
\]

Distance Geometry

A USEFUL IDENTITY (5)

Take the last line of the previous slide and divide through by \( n^2 \):

\[
\frac{1}{n} \left( \sum_{a=1}^{n} d_{a}^{2} + \sum_{\beta=1}^{n} d_{a}^{2} \right) - \frac{1}{n^2} \sum_{a=1}^{n} \sum_{\beta=1}^{n} d_{a\beta}^{2} = x^{(i)T} x^{(i)} + x^{(j)T} x^{(j)}.
\]

This allows us to eliminate \( x^{(i)T} x^{(i)} + x^{(j)T} x^{(j)} \) from our very first equation: \( d_{y}^{2} = x^{(i)T} x^{(i)} - 2x^{(i)T} x^{(j)} + x^{(j)T} x^{(j)} \) which can be rewritten as:

\[
x^{(i)T} x^{(i)} = \frac{1}{2} \left( x^{(i)T} x^{(i)} + x^{(j)T} x^{(j)} - d_{y}^{2} \right)
\]

to give:

\[
x^{(i)T} x^{(j)} = \frac{1}{2} \left[ \frac{1}{n} \left( \sum_{a=1}^{n} d_{a}^{2} + \sum_{\beta=1}^{n} d_{a}^{2} \right) - d_{y}^{2} - \frac{1}{n^2} \sum_{a=1}^{n} \sum_{\beta=1}^{n} d_{a\beta}^{2} \right].
\]
A USEFUL IDENTITY (6)

Another expression for the “double centering formula”:

$$\frac{1}{2} \left[ \frac{1}{n} \left( \sum_{a=1}^{n} d_{aj}^2 + \sum_{b=1}^{n} d_{bj}^2 \right) - d_{ij}^2 - \frac{1}{n^2} \sum_{a=1}^{n} \sum_{b=1}^{n} d_{ab}^2 \right] = x^{(i)} x^{(j)} = G_y.$$  

It can be rewritten as:  

$$G = -\frac{1}{2} H^T D H$$

where:  

$$H = I - \frac{1}{n} \mathbf{1} \mathbf{1}^T$$

and $\mathbf{1}$ is an $n$-dimensional vector with each component equal to 1.

A USEFUL IDENTITY (7)

Showing that:  

$$G = -\frac{1}{2} H^T D H$$

Note:  

$$-H^T D H = \left( \frac{1}{n} \mathbf{1} \mathbf{1}^T - I \right)^T D \left( I - \frac{1}{n} \mathbf{1} \mathbf{1}^T \right) = \frac{1}{n} \left( \mathbf{1} \mathbf{1}^T D + D \mathbf{1} \mathbf{1}^T \right) - D - \frac{1}{n} \mathbf{1} \mathbf{1}^T D \mathbf{1} \mathbf{1}^T.$$  

Also, $D \mathbf{1}$ is an $n$ by 1 column vector with $i^{th}$ entry $\sum_{\beta=1}^{n} d_{\beta i}$.  

so, $D \mathbf{1} \mathbf{1}^T$ is an $n$ by $n$ matrix with $D \mathbf{1}$ as each column.

Similarly, $\mathbf{1} \mathbf{1}^T D$ is a 1 by $n$ row vector with $j^{th}$ entry $\sum_{\alpha=1}^{n} d_{\alpha j}$

and $\mathbf{1} \mathbf{1}^T D$ is an $n$ by $n$ matrix with $\mathbf{1} \mathbf{1}^T D$ as each row.
A USEFUL IDENTITY (8)

- Showing that: $G = -\frac{1}{2} H^T D H$ (continued)
  + The observations on the previous slide explain the two sums:
    $$\sum_{a=1}^{n} d_{a}^2 + \sum_{b=1}^{n} d_{b}^2.$$
  + Note also that:
    $$\bar{T} \bar{D} \bar{I} = \sum_{a=1}^{n} \sum_{b=1}^{n} d_{a}^2. $$
    This is a scalar, so
    $$\bar{T} \bar{D} \bar{I} \bar{T}^T = \left( \sum_{a=1}^{n} \sum_{b=1}^{n} d_{a}^2 \right) \bar{I} \bar{T}^T$$
    which is an $n$ by $n$ matrix with each entry equal to the double sum. This explains the double sum term in the double centering formula.

P1: DERIVING COORDINATES GIVEN G (4)

- The double centering formula allows us to calculate $G$ when given all the squares of the inter-atomic distances (set up in $D$).
  + Then using $G$, we need an algorithm to calculate $X$, the matrix with $x^{(i)}$ stored in column $i$.
  + Before going on, note that $D$ contains $O(n^2)$ entries but our solution $X$ contains only $3n$ values for a 3D space.
    - This means that the entries in $D$ are not arbitrary; they must be consistent with $n$ atoms in a 3D space.
    - The next theorem gives us the necessary constraints.
Theorem:

Given matrix \( D \), there exists a set of points \( \{ x^{(i)} \mid i = 1, 2, \ldots, n \} \) in \( \mathbb{R}^k \) such that
\[
d_{i,j} = \| x^{(i)} - x^{(j)} \| \quad 1 \leq i, j \leq n
\]
if and only if \( G \) is positive semidefinite with rank at most \( k \).
In this case \( G = X^T X \).

The theorem essentially says that an \( X \) solution always exists for any symmetric \( D \) matrix, but if the rank of \( G \) is more than 3, we cannot expect the \( x^{(i)} \) coordinates to be in a 3D space!

This has serious implications for noisy distance data.

To calculate \( X \) we start with the spectral decomposition of \( G \):
\[
G = \sum_{i=1}^{n} \lambda_i u_i^{(m)} (u_i^{(m)})^T \Rightarrow G_{ij} = \sum_{i=1}^{n} \lambda_i u_i^{(m)} (u_j^{(m)})^T.
\]

We eventually want the \( k = 3 \) case but for now it is more informative to consider any fixed integer \( k \in \{1, 2, \ldots, n\} \).
P1: DERIVING COORDINATES GIVEN $G$  

We form the $k$ by $n$ matrix $Y(k)$ defined as:

$$Y(k) = \sqrt{\Lambda(k)} [U(k)]^T \Rightarrow \text{ith column of } Y(k) \text{ is:}$$

$$[y(k)]^{(i)} = [\sqrt{\lambda_1} u_1^{(i)}, \sqrt{\lambda_2} u_2^{(i)}, \ldots, \sqrt{\lambda_k} u_k^{(i)}]^T$$

where $\sqrt{\Lambda(k)}$ is the $k$ by $k$ matrix $\text{diag}(\sqrt{\lambda_1}, \sqrt{\lambda_2}, \ldots, \sqrt{\lambda_k})$ and $U(k)$ is the $n$ by $k$ matrix with column $i$ equal to eigenvector $u_i^{(i)}$.

Then it can be shown that:

$$[y(k)]^{(i)}^T [y(k)]^{(j)} = \sum_{m=1}^{k} \lambda_m u_i^{(m)} u_j^{(m)} \Rightarrow$$

$$G_{ij} - [y(k)]^{(i)}^T [y(k)]^{(j)} = \sum_{m=1}^{n} \lambda_m u_i^{(m)} u_j^{(m)}.$$
When given a complete set of distances that have been subjected to a small amount of noise, we follow the strategy used by Trosset (1998), which reformulates the problem as follows:

\[ \text{minimize } \left\| C - \hat{G} \right\|_F^2 \quad \text{subject to } C \in S^+_n (k) \]

where \( \hat{G} = -\frac{1}{2} H^T \hat{D} H \) and \( S^+_n (k) \) is the set of non-negative \( n \) by \( n \) semidefinite matrices of rank \( k \).

They solve the minimization problem by using the following theorem:

Let \( \hat{\lambda}_1 \geq \hat{\lambda}_2 \geq \ldots \geq \hat{\lambda}_n \) denote the eigenvalues of \( \hat{G} \) with spectral decomposition \( \hat{G} = \hat{U} \hat{\Lambda} \hat{U}^T \) where \( \hat{\Lambda} = \text{diag} \left( \hat{\lambda}_1, \hat{\lambda}_2, \ldots, \hat{\lambda}_n \right) \).

Define:

\[ \bar{\lambda}_i = \max \left\{ 0, \hat{\lambda}_i \right\} \text{ for } i = 1, 2, \ldots, k \quad \text{and} \quad \bar{\lambda}_i = 0 \text{ for } i = k+1, \ldots, n. \]

Let \( \bar{\Lambda} = \text{diag} \left( \bar{\lambda}_1, \bar{\lambda}_2, \ldots, \bar{\lambda}_n \right) \).

Then \( C^* = \hat{U} \bar{\Lambda} \hat{U}^T \) is a global minimiser (and we can use the strategy for P1 with \( C^* \) replacing \( G \)).
SPARSE DISTANCES

- Suppose we have a subset of the inter-atomic distances.
  - For example, NMR gives us all inter-atomic distances characterized as being less than 5 Å.
  - It can be shown that the problem is NP-hard.
  - However, there are heuristics that may be used to get approximate solutions.
  - Various approaches exist.
    - We cover one such strategy called “Geometric Buildup” originally presented in:

GEOMETRIC BUILDUP

- Suppose we are given a subset of inter-atomic distances.
- Recall that we designated this as Problem P3.
  - In geometric buildup we let \( R \) represent the set of atoms with known coordinates.
  - That is, \( R \) is the “determined” set.
  - Initially, \( R \) is empty and the algorithm works by steadily bringing new atoms into \( R \) until the coordinates of all the atoms are determined.
For problem P3 we are given exact distances but only for atoms that are closer than some upper threshold.

**Geometric Buildup** (Wu et al. 2003, 2007, 2008)

+ Find four atoms, not in the same plane, such that **all** inter-atomic distances are known.
+ Using the P1 strategy described earlier, derive the coordinates of all four atoms (so \( R \) starts with 4 atoms).
+ While there are atoms with undetermined positions repeat:
  - Find an atom with undetermined position but with known distances to four other non-coplanar atoms whose positions are known.
  - Determine the position of the undetermined atom using a triangulation strategy.

**Triangulation:**

+ Let \( x^{(k)}_i \) \( i = 1, 2, 3, 4 \) represent the coordinates of four non-coplanar atoms (in \( R \)) with known positions and let \( x^{(j)} \) be the undetermined coordinates of a nearby atom (not in \( R \)) such that all distances between this atom and the four are known. Then:
  \[
  \left\| x^{(k)} \right\|^2 - 2 x^{(k)T} x^{(j)} + \left\| x^{(j)} \right\|^2 = d_{k,j}^2 \quad i = 1, 2, 3, 4
  \]
+ For each \( i = 1, 2, 3 \) we subtract equation \( i \) from equation \( i+1 \):
  \[
  \left\| x^{(k_{i+1})} \right\|^2 - 2 x^{(k_{i+1})T} x^{(j)} + \left\| x^{(j)} \right\|^2 = d_{k_{i+1},j}^2
  \]
  \[
  \left\| x^{(k_i)} \right\|^2 - 2 x^{(k_i)T} x^{(j)} + \left\| x^{(j)} \right\|^2 = d_{k_i,j}^2
  \]
  \[
  \left\| x^{(k_{i+1})} \right\|^2 - \left\| x^{(k_i)} \right\|^2 - 2 \left(x^{(k_{i+1})} - x^{(k_i)}\right)^T x^{(j)} = d_{k_{i+1},j}^2 - d_{k_i,j}^2
  \]
  \( i = 1, 2, 3. \)
P3: SPARSE BUT EXACT DISTANCES (3)

- Rearranging terms:
  \[
  (x^{(k_i)} - x^{(k_j)})^T x^{(j)} = \frac{1}{2} \left( \|x^{(k_i)}\|^2 - \|x^{(k_j)}\|^2 - d_{k_i,j}^2 + d_{k_j,j}^2 \right) \quad i = 1, 2, 3.
  \]

- Set these three equations in matrix form:
  \[
  Bx^{(j)} = c
  \]

where:

\[
B = \begin{bmatrix}
(x^{(k_2)} - x^{(k_1)})^T \\
(x^{(k_3)} - x^{(k_1)})^T \\
(x^{(k_4)} - x^{(k_1)})^T
\end{bmatrix},
\]

\[
c = \frac{1}{2} \begin{bmatrix}
\|x^{(k_2)}\|^2 - \|x^{(k_1)}\|^2 - d_{k_2,j}^2 + d_{k_1,j}^2 \\
\|x^{(k_3)}\|^2 - \|x^{(k_1)}\|^2 - d_{k_3,j}^2 + d_{k_1,j}^2 \\
\|x^{(k_4)}\|^2 - \|x^{(k_1)}\|^2 - d_{k_4,j}^2 + d_{k_1,j}^2
\end{bmatrix}
\]

Distance Geometry

P3: SPARSE BUT EXACT DISTANCES (4)

+ Geometric buildup continued:
  
  We now have the linear system: \( Bx^{(j)} = c \).
  
  Since the \( x^{(k_i)} \) \( i = 1, 2, 3, 4 \) are not coplanar, \( B \) is invertible and we can solve for \( x^{(j)} \) using:
  
  \[ x^{(j)} = B^{-1}c. \]
  
  Then move \( x^{(j)} \) into \( R \).

Repeat the last steps (following the determination of the first four atoms) until all atoms are in \( R \).
Triangulation issues:

- If only three atoms with known positions were used then the position of the undetermined atom would be ambiguous (Fig. 1(a) or Fig. 1(b)?)
- So, we need four atoms but they cannot be coplanar.
- Even if they are “almost coplanar” we have a problem. An ill-conditioned system is very susceptible to noise.

According to the authors:

- “The advantage of using the geometric building algorithm is that it not only is more efficient than the SVD method, but also requires a smaller set of distances and is easier to extend to problems with sparse sets of distances.”
- There is no discussion about numerical stability.

Issues to be considered:

- Consider the first four atoms: They could be at the beginning of a chain or somewhere near the center of a hydrophobic core. Does it matter?
- Intuitively, it would seem that small errors in the inter-atomic distances for the initial atoms of $R$ could propagate throughout the entire solution. Do these small errors magnify as the algorithm progresses?
Recall Problem P4: we are given approximate distances for atoms that are closer than some upper threshold.

+ As long as the distances are positive there is always a solution, but it may reside in a space with dimension higher than 3.
+ If the given distances are true 3D distances with a small amount of error then the first three eigenvalues will be quite different from 0 and the fourth and later eigenvalues will be quite close to 0.
+ Simply ignoring these values (in effect, projecting from a high dimension space down to 3D) can produce an approximate solution typically characterized as “crowded” since contributions to a distance are being ignored.

We now describe a strategy for handling the P4 problem (sparse and noisy distances).

Clique formation:
+ Start by grouping neighbouring atoms to form “cliques”:
  × All inter-atomic distances are known for the atoms in a clique.
  × Recall that we are given all inter-atomic distances less that some threshold (say, τ Angstroms).
  × Each clique will surround a particular atom called the clique center.
P4: MDS & OVERLAPPING CLIQUES

A clique formation heuristic

Suppose atom A is to be a “clique center”.

+ Place all atoms that are within $\tau / 2$ Angstroms of A into an initially empty clique set.
+ Collect all the atoms that are within $\tau$ Angstroms but beyond $\tau / 2$ Angstroms from atom A, and sort them in ascending order with respect to their distance from A.
+ Go through the sorted list formed in the previous step and add an atom to the clique set if the input data includes all inter-atomic distances between that atom and every current member of the clique set.

Choosing clique centers

Clique centers are chosen so that the clique has biological relevance. Each amino acid provides centers for two cliques:

+ A clique centered on the alpha carbon atom:
  + Since the clique center is also a chiral center, we can be sure that the computed coordinates have the appropriate chirality.
+ A clique centered on the carbonyl oxygen atom.
  + This clique overlaps the alpha carbon clique and also includes the hydrogen bonds responsible for helix and strand formation.
Calculating atomic positions

For each clique there is a full set of distance values and so we can use the P2 algorithm discussed earlier.

- Sometimes called an MDS (Multidimensional Scaling) strategy.

The coordinates of all atoms in a clique will be relative to a frame of reference that is only suitable for that clique.

- We need to modify coordinates so that all atoms are in the same frame of reference.

Combining cliques

When cliques overlap, with at least 4 atoms in their intersection, we may combine them:

- This involves a translate and rotate of the second atom set so that both sets have the frame of reference used by the first atom set.
- The atoms in the intersection will define the appropriate translate and rotate operations.
- The intersection atoms cannot be coplanar.
Combining cliques (continued)

Recall that we are using $C^* = \hat{U} \tilde{\Lambda} \hat{U}^\dagger$ where 
$\tilde{\Lambda} = \text{diag}(\tilde{\lambda}_1, \tilde{\lambda}_2, \tilde{\lambda}_3, 0, \ldots, 0)$ and $\tilde{\lambda}_i = \max\{0, \hat{\lambda}_i\}$ $i = 1, 2, 3$.

- Noise in the given distance data will increase the magnitude of $\hat{\lambda}_i$ and thus compromise the ability of $C^*$ to yield the “true” 3D coordinates.
- Consequently, the positions of atoms in the intersection of two cliques will not be precise.

Repeating: Because of noise in the distance data, the computed positions of atoms in the intersection of two cliques are not exact:

- The translate and rotate operations when doing clique combining can still proceed because the super-positioning of atoms in the intersection is done in the least squares sense.
- However, it is still necessary to determine the final positions of atoms in the intersection.
Modifying distance estimates:

Before getting both cliques into the same frame of reference we can try to reduce distance errors by averaging:

+ We can recalculate the squares of distances by using the $C^*$ matrix:

$$ (d_{i,j}^*)^2 = C_{i,i}^* - 2C_{i,j}^* + C_{j,j}^*. $$

Modifying distance estimates (continued):

For any pair of atoms (indexed by $i, j$) within the intersection, each clique will have a different value of $(d_{i,j}^*)^2$.

+ We can try to reduce distance errors by computing an average and then replacing all such distances with the average distance.

+ After this is done, coordinates are computed and one clique can be brought into the frame of reference of the other clique.