# Structured Low Rank Approximation

## Lecture III: Distance Geometry and Protein Structure

Moody T. Chu

North Carolina State University

presented at

XXII School of Computational Mathematics
Numerical Linear Algebra and Its Applications
September 14, 2004

# Syllabus

- Objectives:
  - ◇ To provide some preliminaries.
  - ◇ To treat some mathematics.
  - ◇ To point out some applications.
  - ◇ To describe some algorithms.
- Topics:
  - ◇ Lecture I: Introduction
  - ◇ Lecture II: General Approach
  - ⧫ Lecture III: Distance Geometry and Protein Structure
  - ◇ Lecture IV': Singular Value Assignment with Low Rank Matrices
  - ◇ Lecture V: Nonnegative Matrix Factorization
- Assignments:
  - ◇ Application to some real protein data.
  - ◇ Streamline the code to make it portable.

# Lecture III

# Distance Geometry and Protein Structure

Joint Work with David Chu and Hunter Brown

# Outline

- Introduction
  - ◇ Protein Folding Problem
  - ◇ Euclidean Distance Matrix
- Least Squares Formulation
  - ◇ Approximation Problem
  - ◇ Completion Problem
- Analytic Gradient
- Analytic Hessian
- Numerical Examples

# Introduction

- Fundamental Problem in Distance Geometry:
  - ◇ Given the distance and chirality constrains which define (our state of knowledge of) a mobile molecule,
  - ◇ Find one or more conformations which satisfy them, or else prove that no such conformations exists.
- Applications:
  - ◇ Molecular conformation problems in chemistry.
  - ◇ Multidimensional scaling in behavioral sciences.
  - ◇ Multivariate analysis in statistics.
  - ◇ Remote exploration and sensing, and antenna array processing.
  - ◇ ...

# Protein Folding Problem

- What we know about proteins:
  - ◇ A protein molecule is a connected sequence of amino acid molecules.
  - ◇ There are only twenty amino acids in nature.
    - ▷ Representing each amino acid by a letter from a twenty-letter alphabet, a protein is a string of amino acids linked together like a word.
  - ◇ Most laboratories have the technology to find the ordered sequence of amino acids in a protein.

# Real Issue in Protein Folding

- Merely knowing the long linear chains is not enough.
  - ◇ To function properly, the one-dimensional primary amino acid sequence must fold into a particular three-dimensional conformation called its *tertiary configuration*.
  - ◇ This protein then interacts three-dimensionally with other proteins or other groups of molecules called *substrates* in a manner much like a lock and key arrangement.
- The tertiary structure that mediates how a protein functions.
  - ◇ The final folding configuration determines whether a protein acts as an enzyme in chemical reactions or behaves as part of an antibody in the immune system.
  - ◇ How a protein folds determines how the protein works, which ultimately determines how our bodies work.
  - ◇ Wrongly folded proteins do not behave normally, and their abnormal function can lead to disease.
- Understanding how proteins fold into three-dimensional structures given their primary linear sequence thus becomes an incredibly important task.

# Then What?

- Literature on the various aspects of protein folding is enormous.
  - ◇ Lots of models using a synthesis of knowledge from biology, physics, chemistry, and mathematics.
  - ◇ No clear models.
- Try to see the three dimensional structure.
  - ◇ Biologists use techniques such as x-ray crystallography or molecular dynamics models to peer through the nano-structure.
  - ◇ Noises in the sensing devices or the imperfection of the models often results in indefinite or incomplete images.
- An imperative task:
  - ◇ Retrieve a possible folding structure that is nearest to the observed but possibly inconsistent or imperfect configuration.

# Euclidean Distance Matrix

- Given $n$ particles at locations $\mathbf{p}_1, \ldots, \mathbf{p}_n \in \mathbb{R}^m$,
  - ◇ A complete record of relative spacing between any two of the $n$ particles is called a distance matrix.
  - ◇ The matrix $Q(\mathbf{p}_1, \ldots, \mathbf{p}_n) = [q_{ij}]$ where
  $$q_{ij} = \|\mathbf{p}_i - \mathbf{p}_j\|_2^2, \quad i, j = 1, \ldots, n,$$
  is called a *Euclidean distance matrix*.

# Structures of Eculidean Distance Matrices

- (Schoenberg'35, Young and Householder'38) $Q \in \mathbb{R}^{(n+1)\times(n+1)}$ is a Euclidean distance matrix if and only if $A \in \mathbb{R}^{n\times n}$ defined by

$$a_{ij} := \frac{1}{2}[d_{0i} + d_{0j} - d_{ij}], \quad 1 \le i, j \le n$$

  is positive semidefinite.

  - ◇ $\mathrm{rank}(A)$ is the minimum imbedding dimension, i.e., the lowest dimension of the Euclidean space in which the points reside.
  - ◇ Let $A$ be spectrally decomposed as

$$A = U\Lambda U^\top, \tag{1}$$

    then columns of $C := \Lambda^{1/2}U^\top$ are coordinates for $P_1, \ldots, P_n$.

- (Gower'82) $Q$ is a distance matrix if and only if $\mathbf{x}^\top Q\mathbf{x} \le 0$ for all $\mathbf{x}$ such that $\mathbf{x}^\top\mathbf{1} = 0$.

  - ◇ $Q$ is a distance matrix if and only in $(I - \mathbf{1}\mathbf{s}^\top)Q(I - \mathbf{s}\mathbf{1}^\top)$ is negative semidefinite whenever $\mathbf{s}^\top\mathbf{1} = 1$.

# Rank Structure

- For any $n \geq m+2$, the rank of $Q$ is no greater than $m+2$ and is generically $m+2$.
  - ◇ Write
  $$Q = \left[\|\mathbf{p}_i - \mathbf{p}_j\|^2\right] = \sum_{\ell=1}^{m} \left[(p_{\ell i} - p_{\ell j})^2\right].$$
  - ◇ For each $\ell$,
  $$Q_\ell = \left[(p_{\ell i} - p_{\ell j})^2\right]$$
  is a distance matrix for $n$ points $p_{\ell 1}, \ldots, p_{\ell n}$ on the real line.
  - ◇ Such a matrix can be denoted in a canonical form,
  $$Q_\ell = \begin{bmatrix} 0 & d_1^2 & (d_1+d_2)^2 & (d_1+d_2+d_3)^2 & \cdots \\ d_1^2 & 0 & d_2^2 & (d_2+d_3)^2 & \\ (d_1+d_2)^2 & d_2^2 & 0 & d_3^2 & \\ \vdots & & & & \\ \vdots & & & & \ddots \\ & & & & & 0 \end{bmatrix},$$
  where $d_1, \ldots, d_n$ denote the 1-dimensional coordinates of these points.
  - ◇ $Q_\ell$ can further be reduced by eliminations and scalings to
  $$\begin{bmatrix} -2 & d_1 & 0 & 0 & 0 & 0 & 0 \\ d_1 & 0 & d_1+d_2 & d_2+d_3 & d_3+d_4 & d_4+d_5 & d_5+d_6 \\ 0 & d_1+d_2 & 0 & 0 & 0 & 0 & 0 \\ 0 & d_2+d_3 & 0 & 0 & 0 & 0 & 0 \\ 0 & d_3+d_4 & 0 & 0 & 0 & 0 & 0 \\ 0 & d_4+d_5 & 0 & 0 & 0 & 0 & 0 \\ 0 & d_5+d_6 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

# An Inverse Problem

- Once all the inter-particle distances are in hand, it is a simple task to construct the (3-dimensional) geometric structure.
- Not all the inter-particle distances are needed for the construction.
- Many entries in the matrix provide redundant information.
- Rigid body motions, i.e., rotations, translations, and reflections, of particles in space will not alter the distance matrix.
- Open Question: Characterize when and what partial information of inter-particle distances will be sufficient for the determination of the entire geometric structure.

# Returning to the Scenario

- Let $F \in \mathbb{R}^{n\times n}$ denote an observation of (the squares of) the relative spacing among $n$ particles in $\mathbb{R}^3$.
  - ◇ Some of the spacing has not been measured accurately.
    - ▷ $\operatorname{rank}(F) \geq 5$.
    - ▷ Want to retrieve whatever feasible information out of the matrix $F$ so as to infer a realistic conformation.
  - ◇ Some of the spacing is unobservable.
    - ▷ The matrix $F$ is incomplete.
    - ▷ Want to complete the matrix $F$ so that it becomes a distance matrix.
- Find an Euclidean distance matrix $Q \in \mathbb{R}^{n\times n}$ representing the minimal change of $F$.

# Criteria in Approximation

- Criteria used in characterizing the changes affect the way in which the approximation is formulated.
  - ◇ Low rank approximation techniques (Chu el al'02).
    - ▷ A nonnegative and symmetric matrices of rank 5 is not necessarily a distance matrix.
    - ▷ The distance matrix has more structure that is yet to be exploited.
  - ◇ Alternating projection between two geometric entities (Glunt el al'90).
    - ▷ Slow convergence and possible stagnation.
  - ◇ Semi-definite programming (Alafkin'99).
  - ◇ Limited-memory variable-metric continuation techniques (Moré el al'96).
- Least squares formulation with location vectors $\mathbf{p}_1, \ldots, \mathbf{p}_n$ as parameter.
  - ◇ The resulting approximation of the imperfect $F$ is guaranteed to be a distance matrix.
  - ◇ Not unique under rotation, translation, or reflection.

# Basic Formulation

- Assume initially that all entries of $F$ except the diagonal are to be approximated.
  - ◇ Two special cases later:
    - ▷ The approximation problem where some location vectors of the $n$ particles are known and fixed.
    - ▷ The completion problem where a partial information of spacing among the $n$ particles, but not necessarily from known location vectors, is known and fixed.
- Given $F \in \mathbb{R}^{n\times n}$, minimize the objective function

$$f(\mathbf{p}_1, \ldots \mathbf{p}_n) = \frac{1}{2}\|F - [\langle \mathbf{p}_i - \mathbf{p}_j, \mathbf{p}_i - \mathbf{p}_j\rangle]\|_F^2. \tag{2}$$

  - ◇ $\langle \cdot, \cdot \rangle$ = the usual Euclidean inner product.
  - ◇ $\| \cdot \|_F$ = the Frobenious matrix norm.
- Our contributions:
  - ◇ The general theory is developed in $\mathbb{R}^m$.
  - ◇ Offer a highly organized and effective way of computing the derivative information.

# Analytic Gradient

- Denote discrepancies
$$d_{ij} = f_{ij} - \langle \mathbf{p}_i - \mathbf{p}_j, \mathbf{p}_i - \mathbf{p}_j \rangle, \quad i, j = 1, \dots n. \tag{3}$$
- Consider the space $\mathbb{R}^m \times \dots \times \mathbb{R}^m$ equipped with the product topology.
  - ◇ The Fréchet derivative of $f$ at the point $(\mathbf{p}_1, \dots \mathbf{p}_n)$ acting on an arbitrary *n-fold* vector $(\mathbf{z}_1, \dots, \mathbf{z}_n) \in \mathbb{R}^m \times \dots \times \mathbb{R}^m$ can be represented as the multi-lineal functional
$$f^{'}(\mathbf{p}_1, \dots \mathbf{p}_n).(\mathbf{z}_1, \dots, \mathbf{z}_n) = \sum_{k=1}^{n} \frac{\partial f}{\partial \mathbf{p_k}}(\mathbf{p}_1, \dots \mathbf{p}_n).\mathbf{z}_k \tag{4}$$
    - ▷ $A.\mathbf{b}$ denotes the operator action of $A$ on $\mathbf{b}$.

# Partial Gradient

$$\frac{\partial f}{\partial \mathbf{p}_k} = -4 \sum_{\substack{j=1 \\ j \neq k}}^{n} (\mathbf{p}_k - \mathbf{p}_j) d_{kj}. \tag{5}$$

- Riesz representation:

$$\frac{\partial f}{\partial \mathbf{p_k}}(\mathbf{p}_1, \dots \mathbf{p}_n).\mathbf{z}_k = \langle \frac{\partial f}{\partial \mathbf{p_k}}(\mathbf{p}_1, \dots \mathbf{p}_n), \mathbf{z}_k \rangle.$$

- Carry on:

$$\langle \frac{\partial f}{\partial \mathbf{p_k}}, \mathbf{z}_k \rangle = \langle \frac{\partial}{\partial \mathbf{p}_k} \left( F - [\langle \mathbf{p}_i - \mathbf{p}_j, \mathbf{p}_i - \mathbf{p}_j \rangle] \right) . \mathbf{z}_k, F - [\langle \mathbf{p}_i - \mathbf{p}_j, \mathbf{p}_i - \mathbf{p}_j \rangle] \rangle,$$

- Keep going:

$$\frac{\partial}{\partial \mathbf{p}_k} \left( F - [\langle \mathbf{p}_i - \mathbf{p}_j, \mathbf{p}_i - \mathbf{p}_j \rangle] \right) . \mathbf{z}_k =$$
$$-2 \begin{pmatrix} 0 & \dots & \langle \mathbf{z}_k, \mathbf{p}_k - \mathbf{p}_1 \rangle & & 0 \\ \vdots & & \vdots & & \vdots \\ \langle \mathbf{z}_k, \mathbf{p}_k - \mathbf{p}_1 \rangle & \dots & 0 & \dots & \langle \mathbf{z}_k, \mathbf{p}_k - \mathbf{p}_n \rangle \\ & & \vdots & & \\ 0 & & \langle \mathbf{z}_k, \mathbf{p}_k - \mathbf{p}_n \rangle & & 0 \end{pmatrix}.$$

- Finally,

$$\frac{\partial f}{\partial \mathbf{p_k}}(\mathbf{p}_1, \dots \mathbf{p}_n).\mathbf{z}_k = \langle \mathbf{z}_k, -4 \sum_{\substack{j=1 \\ j \neq k}}^{n} (\mathbf{p}_k - \mathbf{p}_j) d_{kj} \rangle$$

# Gradient of $f$

- Using the product topology again,
$$\nabla f(\mathbf{p}_1, \dots \mathbf{p}_n) = \left( \frac{\partial f}{\partial \mathbf{p}_1}, \dots, \frac{\partial f}{\partial \mathbf{p}_n} \right) \in \mathbb{R}^m \times \dots \times \mathbb{R}^m. \tag{6}$$
- Any numerical method utilizing the gradient information can now be employed to solve the approximation problem.
- Gradient flow:
$$\frac{d\mathbf{p}_k}{dt} = 4 \sum_{\substack{j=1 \\ j \neq k}}^{n} (\mathbf{p}_k - \mathbf{p}_j)(f_{kj} - \langle \mathbf{p}_k - \mathbf{p}_j, \mathbf{p}_k - \mathbf{p}_j \rangle), \quad k = 1, \dots, n$$
moves in the steepest descent direction to reduce the values for the objective function $f$.
  - ◇ A descent flow bounded in a neighborhood of $F$.
  - ◇ Limit point must exist.
  - ◇ A (local) least squares approximation of the given $F$.

# Analytic Hessian

- Will be more effective for finding critical points if the second-derivative information of $f$ is available.
- Consider

$$g : \mathbb{R}^m \times \ldots \times \mathbb{R}^m \to \mathbb{R}^m \times \ldots \times \mathbb{R}^m.$$

  - $g = \nabla f = (g_1, \ldots g_n)$.

$$g_k(\mathbf{p}_1, \ldots \mathbf{p}_n) = \frac{\partial f}{\partial \mathbf{p}_k}(\mathbf{p}_1, \ldots \mathbf{p}_n). \tag{7}$$

- The Jacobian matrix of $g_k$ constitutes precisely the $k$-th row block of the Hessian of $f$.
  - The Hessian can be calculated block by block.

# Partial Jacobian

$$\frac{\partial g_k}{\partial \mathbf{p}_i} = \begin{cases} \sum_{j=1, j\neq k}^{n} \left[-4d_{kj} I_m + 8(\mathbf{p}_k - \mathbf{p}_j)(\mathbf{p}_k - \mathbf{p}_j)^T\right], & \text{if } i = k; \\ 4d_{ki} I_m - 8(\mathbf{p}_k - \mathbf{p}_i)(\mathbf{p}_k - \mathbf{p}_i)^T, & \text{if } i \neq k. \end{cases} \tag{8}$$

- Under the product topology,

$$g_k^{'}(\mathbf{p}_1, \ldots \mathbf{p}_n).(\mathbf{w}_1, \ldots \mathbf{w}_n) = \sum_{i=1}^{n} \frac{\partial g_k}{\partial \mathbf{p}_i}.\mathbf{w}_i.$$

- If $i = k$,

$$\begin{aligned} \frac{\partial g_k}{\partial \mathbf{p}_k}.\mathbf{w}_k &= -4 \sum_{\substack{j=1 \\ j\neq k}}^{n} [\mathbf{w}_k d_{kj} - 2(\mathbf{p}_k - \mathbf{p}_j)\langle \mathbf{w}_k, \mathbf{p}_k - \mathbf{p}_j \rangle] \\ &= \sum_{\substack{j=1 \\ j\neq k}}^{n} \left[-4d_{kj} I_m + 8(\mathbf{p}_k - \mathbf{p}_j)(\mathbf{p}_k - \mathbf{p}_j)^T\right] \mathbf{w}_k. \end{aligned}$$

- If $i \neq k$,

$$\begin{aligned} \frac{\partial g_k}{\partial \mathbf{p}_i}.\mathbf{w}_i &= -4 \left[-\mathbf{w}_i d_{ki} - 2(\mathbf{p}_k - \mathbf{p}_i)\langle -\mathbf{w}_i, \mathbf{p}_k - \mathbf{p}_i \rangle\right] \\ &= \left[4d_{ki} I_m - 8(\mathbf{p}_k - \mathbf{p}_i)(\mathbf{p}_k - \mathbf{p}_i)^T\right] \mathbf{w}_i. \end{aligned}$$

# Modification

- Usually some additional information is available in the geometric conformation.
  - ◇ The x-ray crystal structure of each of the twenty amino acids in nature is known.
  - ◇ Most of the amino acid sequences of our proteins are also known.
  - ◇ Once a certain amino acid is known to present in the protein, a certain block of the matrix $F$ is already predetermined and fixed.
- Least squares formulations should be modified accordingly to reflect this fact.
  - ◇ Derivative information is available in block form.
  - ◇ Convenient for the overall process of assembling essential gradient and the Hessian.

# Approximation with Partially Fixed Locations

- Any rotations, translations, or reflections of a given conformation will produce the same relative spacing and hence the same distance matrix.
  - ◇ To shun rotation, translation, or reflection, $m+1$ positions of these $n$ particles in the embedding space $\mathbb{R}^m$ must be specified and bound as reference points.
  - ◇ Some additional location vectors among $\mathbf{p}_1, \ldots, \mathbf{p}_n$ might be known and fixed beforehand.
- Let $\mathbf{q}$ denote the indices of known location vectors.
  - ◇ Entries $f_{ij}$ of $F$ where both $i, j \in \mathbf{q}$ correspond to the spacing among these known location vectors.
  - ◇ These entries of $F$ should be exact and kept constant.
  - ◇ Derivatives at these points should be zero.
  - ◇ Keep these known position vectors invariant by simply nullifying any derivative information at the corresponding blocks.

# Completion with Partially Fixed $F$

- The matrix $F$ represents a partially specified distance matrix.
  - ◇ The completion problem is different from the approximation problem.
  - ◇ The specified entries in $F$ do not necessarily correspond to any known location vectors, but is required to remain the same throughout the whole matrix completion process.
  - ◇ The specified entries in $F$ must be consistent by themselves to begin with.
  - ◇ Open Question: Determine whether the specified entries in $F$ are consistent so that $F$ indeed can be completed as a distance matrix (Trosett'97).
- Let $\Delta$ denote the index set of those specified entries of $F$.

$$\Delta := \{(i,j) \in \mathbb{Z} \times \mathbb{Z} \,|\, d_{ij} = 0\}\,.$$

- Minimize the same objective function $f(\mathbf{p}_1, \ldots \mathbf{p}_n)$ as before, subject to additional equality constraints

$$\langle \mathbf{p}_i - \mathbf{p}_j, \mathbf{p}_i - \mathbf{p}_j \rangle = f_{ij} \quad \text{for all } (i,j) \in \Delta. \tag{9}$$

  - ◇ The completion problem is cast as an equality constrained optimization problem.

# Example 1

- A knot $\mathbb{R}^3$.

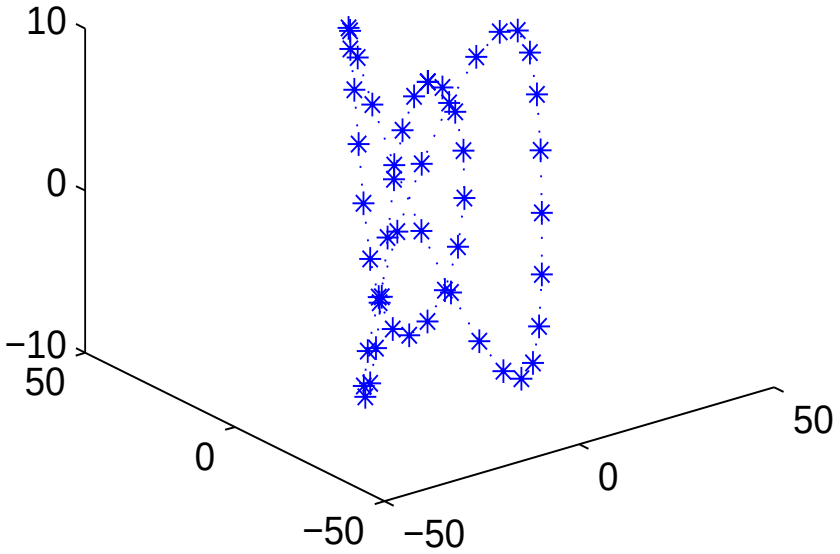
$$\begin{cases} x = -10\cos(t) - 2\cos(5t) + 15\sin(2t); \\ y = -15\cos(2t) + 10\sin(t) - 2\sin(5t); \qquad 0 \le t \le 2\pi \\ z = 10\cos(3t); \end{cases}$$

- Simulation:
  - ◇ Represent the knot by $n$ discrete points.
  - ◇ Use the points to define the true $n \times n$ distance matrix $Q = [q_{ij}]$.
  - ◇ Let $\mathbf{q}$ denote the indices of known location vectors.
    - ▷ $\mathbf{q}$ must contain at least four points.
  - ◇ Perturb $Q$ to simulate $F$.

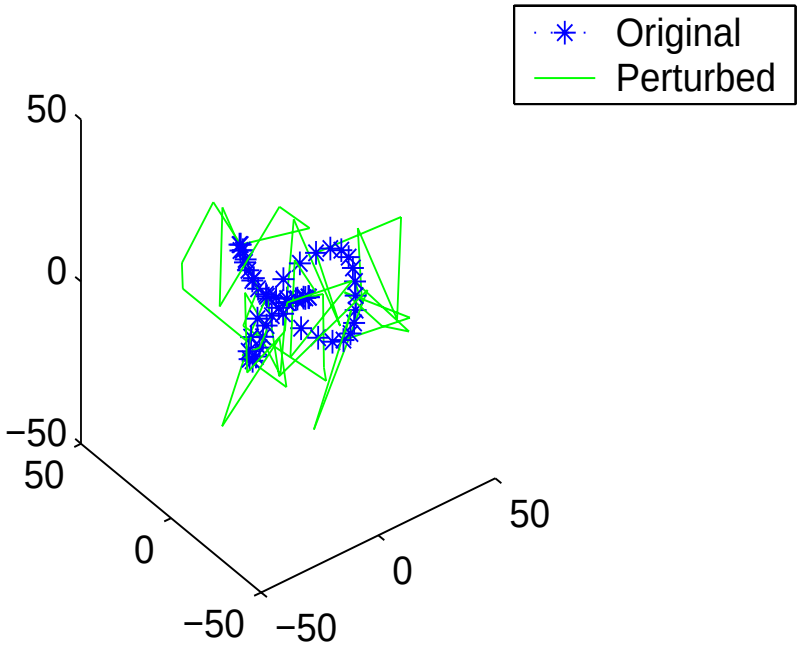
$$f_{ij} = f_{ji} = \begin{cases} 0, & \text{if } i = j, \\ q_{ij}, & \text{if both } i \text{ and } j \text{ are in } \mathbf{q}, \\ \left(\sqrt{q_{ij}} + \sigma * randn(1)\right)^2, & \text{if either } i \text{ or } j \text{ is not in } \mathbf{q}. \end{cases}$$

    - ▷ $\sigma$ is an indicator of how far $F$ is away from a distance matrix.
    - ▷ $\sigma = 2$ would bury $Q$ in a significant amount of random noise.
  - ◇ Use existing routine FMINUNC in the Optimization Toolbox, Version 2.2, of MATLAB.
  - ◇ Keep location vectors in by $\mathbf{q}$ fixed throughout the iteration.
  - ◇ Perturb each entry of the true location vectors by an additive noise with uniform distribution over [-20,20].
- Numerical results:
  - ◇ Objective values have been reduced from an order of $10^9$ to $10^6$, indicating that $F$ is far from being a distance matrix.
  - ◇ Remarkable likeness between the recovered and the original configurations.

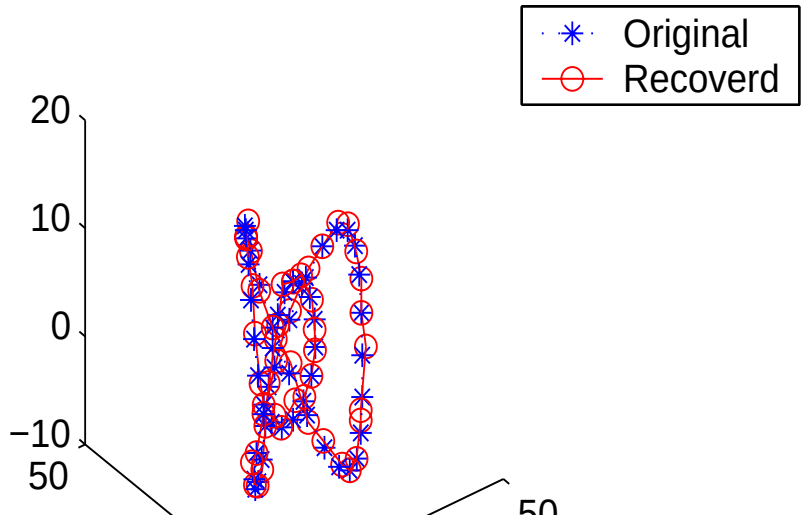
Original Configuration



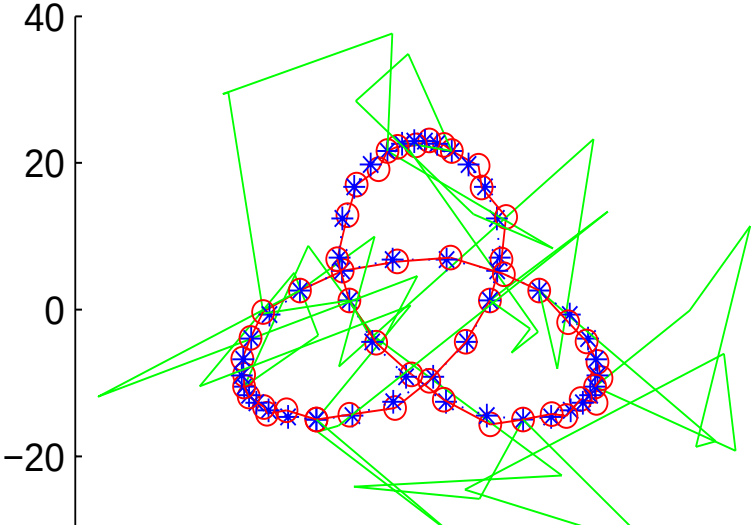

Original and Perturbed Configurations




Original and Recovered Configurations




2D View of Orignial, Perturbed, and Recovered Configurations



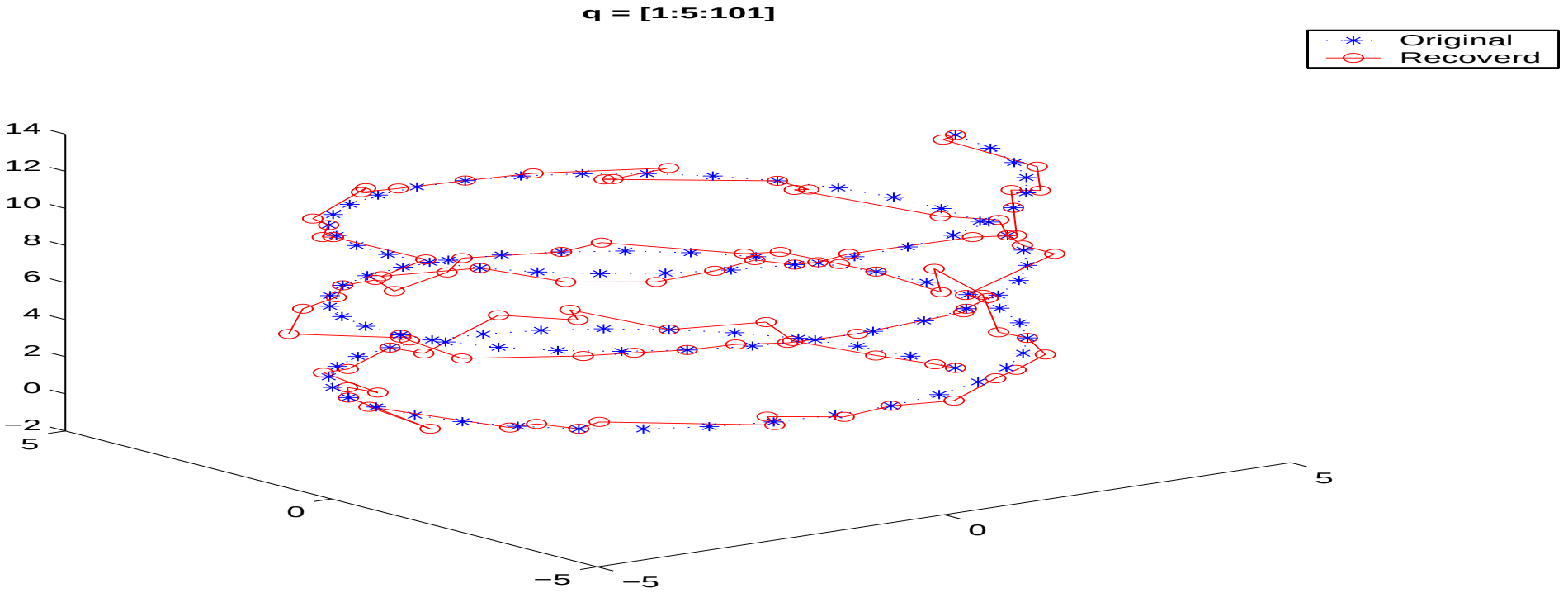
# Example 2.

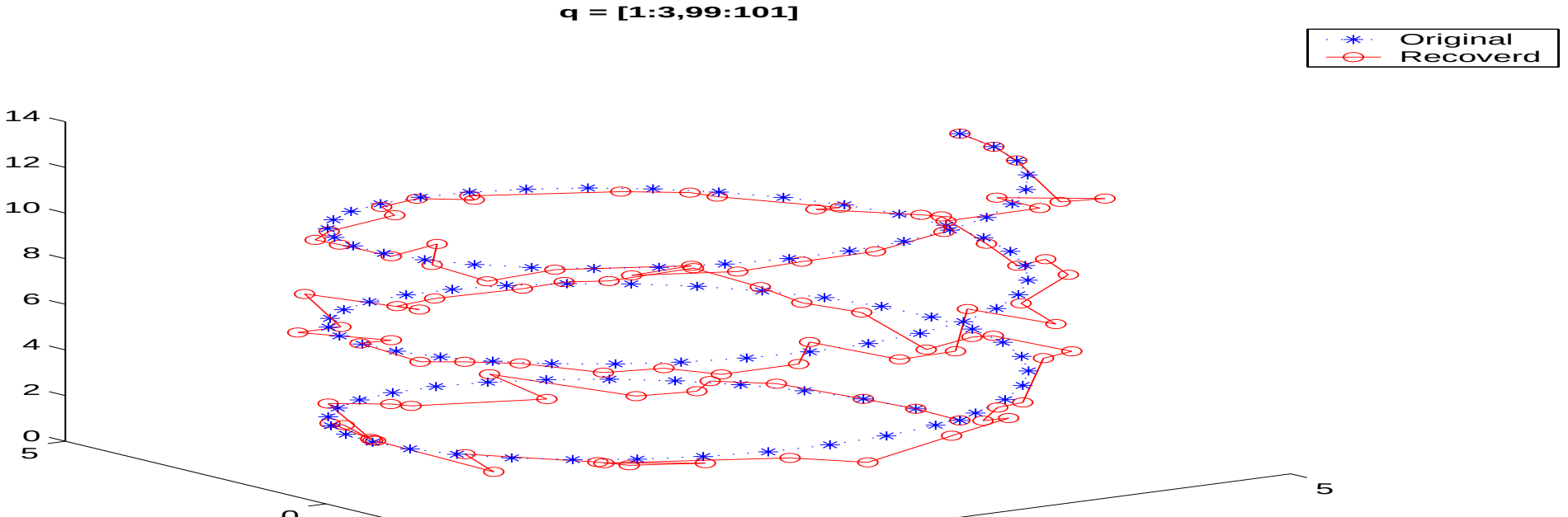
---

- A helix

$$\begin{cases} x = 4\cos(3t); \\ y = 4\sin(3t); \quad 0 \le t \le 2\pi; \\ z = 2t. \end{cases}$$

- Numerical results:
  - ◇ $n = 101$ and $\mathbf{q} = 1 : 5 : 101$.
    - ▷ Not a smooth solution.
    - ▷ Recapture the helix feature from a fairly deviate initial guess and a fraudulent $F$.
    - ▷ Pre-shaped?
  - ◇ $\mathbf{q} = [1 : 3, 99 : 101]$.
    - ▷ Only the first and the last three location vectors are fixed.
    - ▷ Deviate from the helix further.
    - ▷ Spirality is evident.


q = [1:5:101]
Original
Recoverd
14
12
10
8
6
4
2
0
−2
5
0
−5
−5
0
5


q = [1:3,99:101]
Original
Recoverd
14
12
10
8
6
4
2
0
5
5

# Example 3

- A 2-dimensional pretzel:

$$\begin{cases} x = t + 3\sin(2t); \\ y = t + 2\sin(3t); \quad -2\pi \le t \le 2\pi. \end{cases}$$

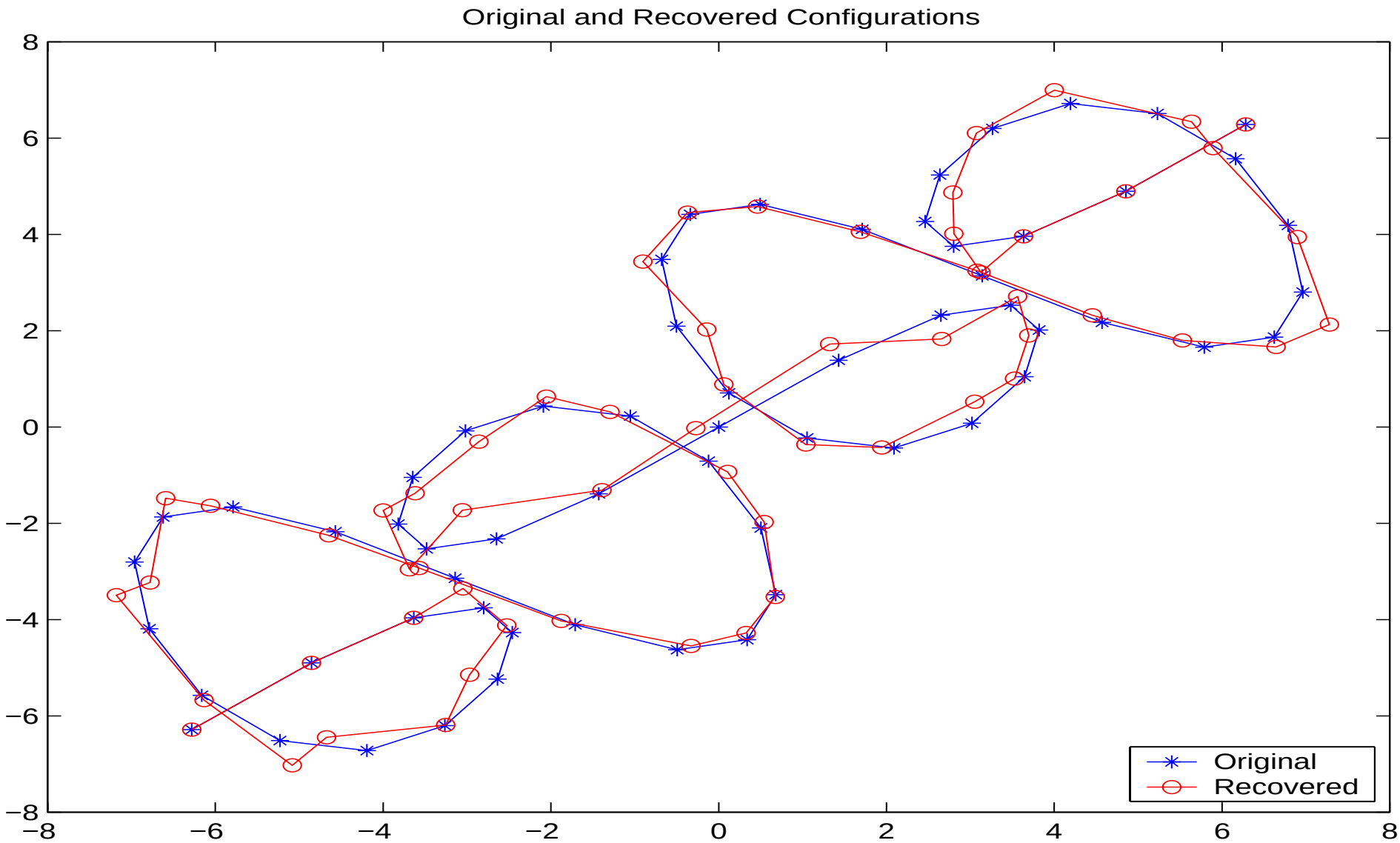
  - ◇ Has many critical turns which are close to each other.
  - ◇ A more careful observation of the spacing among the location vectors is necessary.
    - ▷ To affect this scenario, we assume a smaller standard deviation $\sigma = 1$ of noise in the simulation.

- Simulation:

  - ◇ $n = 61$.
  - ◇ $\mathbf{q} = [1:3, 59:61]$.
  - ◇ Moderately wild initial guess.

- Numerical results:

  - ◇ Able to pick up the folds.
  - ◇ Smaller $\sigma$ or more known position would ease the difficulties.


Original and Recovered Configurations
8
6
4
2
0
−2
−4
−6
−8
−8
−6
−4
−2
0
2
4
6
8
Original
Recovered

# Example 4

- Complete the $6 \times 6$ partially specified matrix

$$F = \begin{bmatrix} 0 & 3 & 4 & 3 & 4 & 3 \\ 3 & 0 & 1 & x_1 & 5 & x_3 \\ 4 & 1 & 0 & 5 & x_2 & 5 \\ 3 & x_1 & 5 & 0 & 1 & x_4 \\ 4 & 5 & x_2 & 1 & 0 & 5 \\ 3 & x_3 & 5 & x_4 & 5 & 0 \end{bmatrix}$$

  with values of $x_i$, $i = 1, \ldots, 4$.

  - ◇ Look for six location vectors in $\mathbb{R}^m$.
  - ◇ Not sure about the dimension $m$ of the embedding space.
  - ◇ The first $3 \times 3$ principal submatrix is completely known, suggesting that three location vectors could have been self-determined.

- If $m = 2$,

  - ◇ Use $\mathbf{p}_1 = (0, 0)$, $\mathbf{p}_2 = (\sqrt{3}, 0)$, and $\mathbf{p}_3 = (\sqrt{3}, 1)$ as reference points in $\mathbb{R}^2$.
  - ◇ Eight equality constraints in the form of (9) for the remaining three location vectors.
  - ◇ Unknown location vectors $\mathbf{p}_j$, $j = 4, 5, 6$, constitute only six unknowns.
  - ◇ No feasible solution at all in $\mathbb{R}^2$.

- If $m = 3$,
  - ◇ Embed $\mathbf{p}_i$, $i = 1, 2, 3$, in $\mathbb{R}^3$ by adding zeros to their third components.
  - ◇ Employ an existing routine FMINCON in the Optimization Toolbox of MATLAB by supplying the equality constraints (9).
- Numerical results.

| $x_1$ | $x_2$ | $x_3$ | $x_4$ |
|---|---|---|---|
| 6.6883 | 3.9512 | 2.0187 | 7.3255 |
| 1.7434 | 9.1772 | 2.2007 | 2.2006 |
| 2.2800 | 9.4157 | 2.3913 | 4.7487 |
| 2.7971 | 5.7203 | 7.2315 | 7.2315 |
| 2.2723 | 9.4208 | 2.3964 | 4.7398 |

Table 1: Examples of entries for completed distance matrix.

| $\mathbf{p}_1$ | $\mathbf{p}_2$ | $\mathbf{p}_3$ | $\mathbf{p}_4$ | $\mathbf{p}_5$ | $\mathbf{p}_6$ |
|---|---|---|---|---|---|
| 0 | 1.7321 | 1.7321 | 0.9014 | 0.5774 | 1.1359 |
| 0 | 0 | 1.0000 | -0.5613 | -0.4223 | -0.9675 |
| 0 | 0 | 0 | 1.3335 | 1.7980 | -0.2711 |
| 0 | 0 | 0 | -0.3067 | 0.5057 | 0.8367 |

Table 2: Example of location vectors in $\mathbb{R}^4$ for completed distance matrix.

# Conclusion

- There are several algorithms available for solving the distance matrix approximation problem.
  - ◇ We cast the problem under a least squares approximation in terms of the location vectors directly and propose using conventional large-scale optimization techniques instead.
- We manage the resulting complexity by organizing the gradient and Hessian information in block forms.
  - ◇ The matrix calculus makes it particularly easy to assemble the derivatives for existing software packages when some locations vectors are known and fixed.
- Numerical experiments seem to suggest that the conventional methods are efficient and robust in the reconstruction.