Geometric Buildup Methods for Protein Structure Determination

Zhenli Sheng

Institute of Computational Mathematics and Scientific/Engineering Computing,
Chinese Academy of Sciences

June 12, 2012

Outline

Outline

1. Problem Introduction

Outline

1. Problem Introduction

2. Related Works

Outline

1. Problem Introduction

2. Related Works

3. Distributed Geometric Buildup Method

Outline

1. Problem Introduction
2. Related Works
3. Distributed Geometric Buildup Method
4. Two New Models

Outline

1. Problem Introduction
2. Related Works
3. Distributed Geometric Buildup Method
4. Two New Models
5. Conclusions and Future work

Distance Geometry(DG) Problem

Find the coordinate vectors $x_1, x_2, \ldots, x_n$ that satisfy several given distances between them.

Distance Geometry(DG) Problem

Find the coordinate vectors $x_1, x_2, \ldots, x_n$ that satisfy several given distances between them.

- data given:
  - exact distances (error-free)
  - inexact distances (with noises)
  - distance bounds

Distance Geometry(DG) Problem

Find the coordinate vectors $x_1, x_2, \ldots, x_n$ that satisfy several given distances between them.

- data given:
  - exact distances (error-free)
  - inexact distances (with noises)
  - distance bounds

- applications:
  - graph realization
  - protein structure determination (3D)
  - sensor network localization (2D)
  - ...

Related works

Related works

- Matrix Decomposition Method (Blumenthal 1953, Torgerson 1958)

Related works

- Matrix Decomposition Method (Blumenthal 1953, Torgerson 1958)
- The Embedding Algorithm (Crippen, Havel 1988)

Related works

- Matrix Decomposition Method (Blumenthal 1953, Torgerson 1958)
- The Embedding Algorithm (Crippen, Havel 1988)
- Global Smoothing Algorithm (Moré, Wu 1997)

Related works

- Matrix Decomposition Method (Blumenthal 1953, Torgerson 1958)
- The Embedding Algorithm (Crippen, Havel 1988)
- Global Smoothing Algorithm (Moré, Wu 1997)
- Geometric Buildup Method (Dong, Wu 2002)

Related works

- Matrix Decomposition Method (Blumenthal 1953, Torgerson 1958)
- The Embedding Algorithm (Crippen, Havel 1988)
- Global Smoothing Algorithm (Moré, Wu 1997)
- Geometric Buildup Method (Dong, Wu 2002)
- SDP Relaxiation Method (Ye, et al., 2006)

Related works

- Matrix Decomposition Method (Blumenthal 1953, Torgerson 1958)
- The Embedding Algorithm (Crippen, Havel 1988)
- Global Smoothing Algorithm (Moré, Wu 1997)
- Geometric Buildup Method (Dong, Wu 2002)
- SDP Relaxiation Method (Ye, et al., 2006)
- …

Matrix Decomposition Method

DG problem with full set of exact distances

Given a full set of distances, $d_{i,j} = \|x_i - x_j\|, \quad i,j = 1,2,\ldots,n.$

Matrix Decomposition Method

DG problem with full set of exact distances

Given a full set of distances, $d_{i,j} = \|x_i - x_j\|, \quad i,j = 1,2,\ldots,n.$

- Set $x_n = (0,0,0)^{\mathrm{T}}$, we have

$$\begin{aligned} d_{i,j}^2 &= \|x_i - x_j\|^2 \\ &= \|x_i\|^2 - 2x_i^{\mathrm{T}} x_j + \|x_j\|^2 \\ &= d_{i,n}^2 - 2x_i^{\mathrm{T}} x_j + d_{j,n}^2 \qquad i,j = 1,2,\ldots,n-1 \end{aligned} \tag{1}$$

Matrix Decomposition Method

DG problem with full set of exact distances

Given a full set of distances, $d_{i,j} = \|x_i - x_j\|, \quad i,j = 1,2,\ldots,n.$

- Set $x_n = (0,0,0)^{\mathrm{T}}$, we have

$$\begin{aligned} d_{i,j}^2 &= \|x_i - x_j\|^2 \\ &= \|x_i\|^2 - 2x_i^{\mathrm{T}} x_j + \|x_j\|^2 \\ &= d_{i,n}^2 - 2x_i^{\mathrm{T}} x_j + d_{j,n}^2 \qquad i,j = 1,2,\ldots,n-1 \end{aligned} \tag{1}$$

- Define $X = (x_1, x_2, \ldots, x_n)^{\mathrm{T}}$ and
$D = \{(d_{i,n}^2 - d_{i,j}^2 + d_{j,n}^2)/2 : i,j = 1,2,\ldots,n-1\}$, (1) $\Rightarrow XX^{\mathrm{T}} = D$.

Matrix Decomposition Method

DG problem with full set of exact distances

Given a full set of distances, $d_{i,j} = \|x_i - x_j\|, \quad i, j = 1, 2, \ldots, n.$

- Set $x_n = (0, 0, 0)^{\mathrm{T}}$, we have

$$\begin{aligned} d_{i,j}^2 &= \|x_i - x_j\|^2 \\ &= \|x_i\|^2 - 2x_i^{\mathrm{T}} x_j + \|x_j\|^2 \\ &= d_{i,n}^2 - 2x_i^{\mathrm{T}} x_j + d_{j,n}^2 \qquad i, j = 1, 2, \ldots, n-1 \end{aligned} \tag{1}$$

- Define $X = (x_1, x_2, \ldots, x_n)^{\mathrm{T}}$ and $D = \{(d_{i,n}^2 - d_{i,j}^2 + d_{j,n}^2)/2 : i, j = 1, 2, \ldots, n-1\}$, (1) $\Rightarrow$ $XX^{\mathrm{T}} = D$.

- Let $D = U \Sigma U^{\mathrm{T}}$, and $V = U(:, 1:3), \Lambda = \Sigma(1:3, 1:3)$. Then $X = V \Lambda^{1/2}$ solves the problem. [Eckart and Young 1936]

Geometric Buildup Method

Geometric Buildup Method

- Find four atoms to form a base
  - determine their coordinates to remove the possible translation and rotation/reflection

Geometric Buildup Method

- Find four atoms to form a base
  - determine their coordinates to remove the possible translation and rotation/reflection

- Determine atoms one by one
  - at least four distances from the undetermined atom to determined atoms are known

Geometric Buildup Method

- Find four atoms to form a base
  - determine their coordinates to remove the possible translation and rotation/reflection

- Determine atoms one by one
  - at least four distances from the undetermined atom to determined atoms are known

Zachary Voller, Zhijun Wu(2012), Distance Geometry Methods for Protein Structure Determination.

Determine one unknown atom

Given four determined atoms x_1, x_2, x_3 and x_4, which $x_i = (x_{i1}, x_{i2}, x_{i3})^{\mathrm{T}}$ are known, and four exact distances.

Determine one unknown atom

Given four determined atoms x_1, x_2, x_3 and x_4, which $x_i = (x_{i1}, x_{i2}, x_{i3})^{\mathrm{T}}$ are known, and four exact distances.

- $d_{i,j}^2 = \|x_i\|^2 - 2x_i^{\mathrm{T}} x_j + \|x_j\|^2 \qquad i = 1, 2, 3, 4.$

Determine one unknown atom

Given four determined atoms x_1, x_2, x_3 and x_4, which $x_i = (x_{i1}, x_{i2}, x_{i3})^{\mathrm{T}}$ are known, and four exact distances.

- $d_{i,j}^2 = \|x_i\|^2 - 2x_i^{\mathrm{T}} x_j + \|x_j\|^2 \qquad i = 1, 2, 3, 4.$
- $\Rightarrow Ax_j = b$,

  where $A = 2 \begin{pmatrix} x_{11} - x_{21} & x_{12} - x_{22} & x_{13} - x_{23} \\ x_{21} - x_{31} & x_{22} - x_{32} & x_{23} - x_{33} \\ x_{31} - x_{41} & x_{32} - x_{42} & x_{33} - x_{43} \end{pmatrix}$

  and $b = \begin{pmatrix} (\|x_1\|^2 - \|x_2\|^2) - (d_{1,j}^2 - d_{2,j}^2) \\ (\|x_2\|^2 - \|x_3\|^2) - (d_{2,j}^2 - d_{3,j}^2) \\ (\|x_3\|^2 - \|x_4\|^2) - (d_{3,j}^2 - d_{4,j}^2) \end{pmatrix}.$

Determine one unknown atom

Given four determined atoms x_1, x_2, x_3 and x_4, which $x_i = (x_{i1}, x_{i2}, x_{i3})^{\mathrm{T}}$ are known, and four exact distances.

- $d_{i,j}^2 = \|x_i\|^2 - 2x_i^{\mathrm{T}} x_j + \|x_j\|^2 \qquad i = 1, 2, 3, 4.$
- $\Rightarrow Ax_j = b$,

  where $A = 2 \begin{pmatrix} x_{11} - x_{21} & x_{12} - x_{22} & x_{13} - x_{23} \\ x_{21} - x_{31} & x_{22} - x_{32} & x_{23} - x_{33} \\ x_{31} - x_{41} & x_{32} - x_{42} & x_{33} - x_{43} \end{pmatrix}$

  and $b = \begin{pmatrix} (\|x_1\|^2 - \|x_2\|^2) - (d_{1,j}^2 - d_{2,j}^2) \\ (\|x_2\|^2 - \|x_3\|^2) - (d_{2,j}^2 - d_{3,j}^2) \\ (\|x_3\|^2 - \|x_4\|^2) - (d_{3,j}^2 - d_{4,j}^2) \end{pmatrix}.$

- Inexact distances?

Linear and Nonlinear Least-squares Approximation

DG problem with inexact distances

Suppose l distances between the unknown atom to the determined atoms are known.

Linear and Nonlinear Least-squares Approximation

DG problem with inexact distances

Suppose l distances between the unknown atom to the determined atoms are known.

- linear least-squares
  - use only the l distances
  - $\min \|b - Ax_j\|$

Linear and Nonlinear Least-squares Approximation

DG problem with inexact distances

Suppose l distances between the unknown atom to the determined atoms are known.

- linear least-squares
  - use only the l distances
  - $\min \|b - Ax_j\|$
- nonlinear least-squares
  - use all the distances among the $l+1$ atoms
  - solve a matrix decomposition problem
  - move the same points in two different reference system coincide

Atilla Sit, Zhijun Wu and Ya-xiang Yuan(2009), A geometric buildup algorithm for the solution of the distance geometry problem using least-squares approximation.

Laplacian Matrix

Given a graph (V, E), define its Laplacian matrix by L, whose entries $l_{i,j}$ are given by

Laplacian Matrix

Given a graph (V, E), define its Laplacian matrix by L, whose entries $l_{i,j}$ are given by

- $$l_{i,j} = \begin{cases} deg(v_i) & \text{if } i = j, & \rightarrow -sum(L(i,:)) \\ -1 & \text{if } (i,j) \in E, & \rightarrow -exp(-d_{i,j}^2/2) \\ 0 & \text{otherwise.} & \end{cases}$$

Laplacian Matrix

Given a graph (V, E), define its Laplacian matrix by L, whose entries $l_{i,j}$ are given by

- $$l_{i,j} = \begin{cases} deg(v_i) & \text{if } i = j, & \rightarrow -sum(L(i,:)) \\ -1 & \text{if } (i,j) \in E, & \rightarrow -exp(-d_{i,j}^2/2) \\ 0 & \text{otherwise.} & \end{cases}$$

- $L = D - A$, where D is degree matrix, and A is its adjacency matrix.

Laplacian Matrix

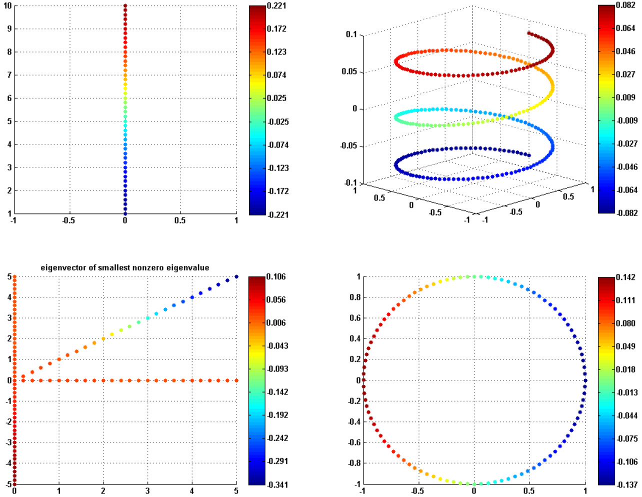
Given a graph (V, E), define its Laplacian matrix by L, whose entries $l_{i,j}$ are given by

- $$l_{i,j} = \begin{cases} deg(v_i) & \text{if } i = j, & \rightarrow -sum(L(i,:)) \\ -1 & \text{if } (i,j) \in E, & \rightarrow -exp(-d_{i,j}^2/2) \\ 0 & \text{otherwise.} & \end{cases}$$

- $L = D - A$, where D is degree matrix, and A is its adjacency matrix.

- Properties:
  - L is always positive-semidefinite.
  - 0 is always its eigenvalue and its corresponding eigenvector is $(1, 1, \ldots, 1)^{\mathrm{T}}$.
  - The number of times 0 appears as an eigenvalue in the Laplacian is the number of connected components in the graph.

Eigenvector of smallest nonzero eigenvalue

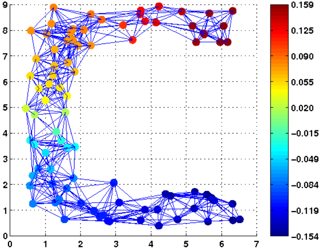


eigenvector of smallest nonzero eigenvalue

a Conjecture

Conjecture

Given a graph (V, E), its Laplacian matrix is defined as before, then the eigenvector of the smallest nonzero eigenvalue, which can be viewed as a function of the vertexes, monotonically decrease or increase along the main trend/direction of the graph.



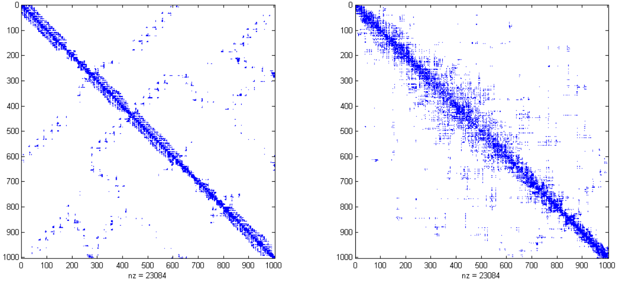

0
100
200
300
400
500
600
700
800
900
1000
nz = 23084
0
100
200
300
400
500
600
700
800
900
1000
nz = 23084

Figure : 1AX8

Implement Details

Implement Details

- computational order:

IAX8, 1003 atoms, cutoff=5Å, 2.3% exact		
order	RmsdErr	CPU time (s)
original	7.395926e+000	1.258295
greedy	9.281000e-005	2.346640
Laplacian	9.281000e-005	1.999627
random	7.372537e-007	4.980335
	1.726266e-007	2.946320
	3.858138e-003	2.988656
	2.437341e-008	3.517570
	1.223379e-005	4.757714
	1.169260e-003	5.339559
	1.399711e-006	2.478225
	1.771925e-005	4.957635
	9.559394e-009	2.750663
	8.637780e-007	5.196890

- computational order:

1MQQ, 5681 atoms, cutoff=6Å, 0.75%, exact		
order	RmsdErr	CPU time (s)
original	1.130061e+001	11.262673
greedy	4.310119e-003	53.904010
Laplacian	5.039401e-005	135.032459
random	5.315594e-003	23.218307
	1.612265e-002	67.657580
	2.928411e-004	142.834237
	2.262457e-007	29.632780
	3.823293e-006	70.646800
	3.165929e-003	140.470924
	8.535506e-001	254.812797
	6.665072e-005	214.758550
	4.334586e-001	141.894475
	2.888975e-001	23.932108

PDB ID	No.	greedy order				Laplacian order		
		RmsdErr	CPU time	NumDet		RmsdErr	CPU time	NumD
1PTQ	402	1.15e-012	1.22e+000	402		8.97e-013	4.98e-001	402
1HOE	558	1.50e-012	8.09e-001	558		1.08e-011	2.17e+000	558
1LFB	641	7.59e-010	9.10e-001	641		1.75e-010	1.51e+000	641
1PHT	811	1.61e-011	1.23e+000	811		1.67e-013	8.41e+001	4
1POA	914	6.17e-010	1.43e+000	914		3.31e-011	1.59e+000	914
1AX8	1003	1.24e-011	1.68e+000	1003		4.87e-007	3.62e+000	1003
1F39	1534	2.32e-006	3.86e+000	1534		4.03e-014	2.93e+002	1534
1RGS	461	2.61e-014	2.11e-001	4		3.33e-014	2.14e+000	4
1KDH	2846	7.15e-004	8.56e+000	2846		1.58e-001	5.17e+000	2846
1BPM	3671	4.45e-005	9.03e+000	3671		9.45e-013	9.92e+002	4
1RHJ	3740	3.47e-008	1.07e+001	3740		1.00e-006	1.19e+001	3740
1HQQ	3944	4.77e-006	1.22e+001	3944		2.76e+000	7.43e+000	3944
1TOA	4292	2.35e+001	2.79e+001	4292		1.09e-001	8.93e+000	4292
1MQQ	5681	4.31e-003	4.98e+001	5681		1.55e-002	5.48e+001	5681

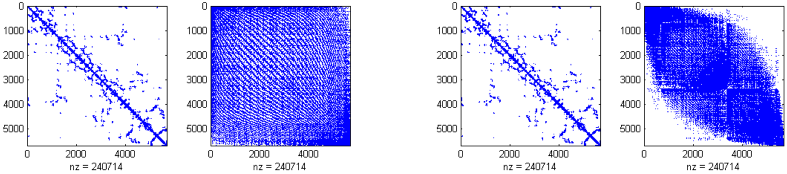


Figure : 1MQQ, greedy, theoretical VS. real order

Some other problems

Some other problems

- not enough bases

Some other problems

- not enough bases
- bad condition number: say, cond$(A^{\mathrm{T}}A) > 10^6$

  $\Leftarrow$ bases almost in the same plane!

Some other problems

- not enough bases
- bad condition number: say, cond$(A^{\mathrm{T}}A) > 10^6$

  $\Leftarrow$ bases almost in the same plane!
- $\rightsquigarrow$ move to last

Motivation

1MQQ, 5681 atoms, cutoff=6Å, 0.75%, exact distances

Itr	RmsdErr	Itr	RmsdErr
300	1.020647e-012	3000	1.186818e-004
600	1.805403e-010	3300	3.477342e-004
900	2.059775e-007	3600	1.953754e-003
1200	6.691896e-007	3900	1.973875e-003
1500	3.358551e-006	4200	2.162615e-003
1800	4.677271e-006	4500	2.231129e-003
2100	7.869284e-006	4800	2.790680e-003
2400	2.062700e-005	5100	3.084867e-003
2700	6.988388e-005	5400	4.277911e-003

Motivation

1MQQ, 5681 atoms, cutoff=6Å, 0.75%, exact distances

- Error Accumulation!

Itr	RmsdErr	Itr	RmsdErr
300	1.020647e-012	3000	1.186818e-004
600	1.805403e-010	3300	3.477342e-004
900	2.059775e-007	3600	1.953754e-003
1200	6.691896e-007	3900	1.973875e-003
1500	3.358551e-006	4200	2.162615e-003
1800	4.677271e-006	4500	2.231129e-003
2100	7.869284e-006	4800	2.790680e-003
2400	2.062700e-005	5100	3.084867e-003
2700	6.988388e-005	5400	4.277911e-003

Divide and Conquer

Divide and Conquer

- Divide into small patches

Divide and Conquer

- Divide into small patches
- Geometric Buildup at each patch

Divide and Conquer

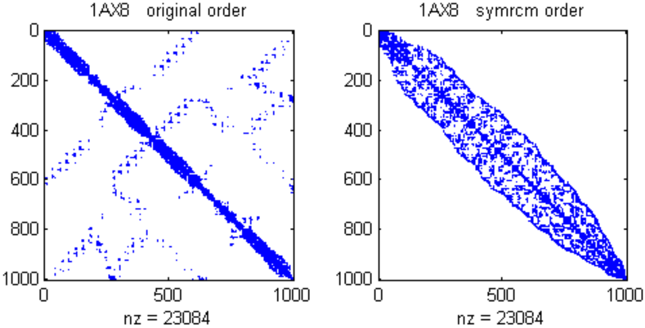
- Divide into small patches
- Geometric Buildup at each patch
- Stitch together

How to divide

How to divide

- symrcm: minimize the bandwidth

  Pratik Biswas, Kim-Chuan Toh and Yinyu Ye(2007), A Distributed SDP Approach for Large-scale Noisy Anchor-free Graph Realization with Application to Molecular Conformation.



How to stitch

Given two 3D point sets $\{p_i\}$ and $\{q_i\}$, $i = 1, 2, \ldots, k$.

How to stitch

Given two 3D point sets $\{p_i\}$ and $\{q_i\}$, $i = 1, 2, \ldots, k$.

- $$\begin{aligned} \min_{R,T} \quad & \sum_{i=1}^{k} \|p_i - (Rq_i + T)\|_2^2 \\ \text{s.t.} \quad & R^{\mathrm{T}} R = I. \end{aligned} \tag{2}$$

How to stitch

Given two 3D point sets $\{p_i\}$ and $\{q_i\}$, $i = 1, 2, \ldots, k$.

- $$\begin{aligned} \min_{R,T} \quad & \sum_{i=1}^{k} \|p_i - (Rq_i + T)\|_2^2 \\ \text{s.t.} \quad & R^{\mathrm{T}} R = I. \end{aligned} \tag{2}$$
- T: make their geometric center coincide

How to stitch

Given two 3D point sets $\{p_i\}$ and $\{q_i\}$, $i = 1, 2, \ldots, k$.

- $$\begin{aligned} \min_{R,T} \quad & \sum_{i=1}^{k} \|p_i - (Rq_i + T)\|_2^2 \\ \text{s.t.} \quad & R^{\mathrm{T}} R = I. \end{aligned} \tag{2}$$
- T: make their geometric center coincide
- R:

$$\begin{aligned} \min_{R} \quad & \|P - RQ\|_F^2 \\ \text{s.t.} \quad & R^{\mathrm{T}} R = I. \end{aligned} \tag{3}$$

Let $C = PQ^{\mathrm{T}}$, and $C = U\Sigma V^{\mathrm{T}}$, then $R = VU^{\mathrm{T}}$ solves (3). [Matrix Computation, Golub]

How to stitch

Given two 3D point sets $\{p_i\}$ and $\{q_i\}$, $i = 1, 2, \ldots, k$.

- $$\begin{aligned} \min_{R,T} \quad & \sum_{i=1}^{k} \|p_i - (Rq_i + T)\|_2^2 \\ \text{s.t.} \quad & R^{\mathrm{T}} R = I. \end{aligned} \tag{2}$$
- T: make their geometric center coincide
- R:

  $$\begin{aligned} \min_{R} \quad & \|P - RQ\|_F^2 \\ \text{s.t.} \quad & R^{\mathrm{T}} R = I. \end{aligned} \tag{3}$$

  Let $C = PQ^{\mathrm{T}}$, and $C = U\Sigma V^{\mathrm{T}}$, then $R = VU^{\mathrm{T}}$ solves (3). [Matrix Computation, Golub]
- Remark: a fundamental problem in Machine Intelligence and Optical Science.

Numerical experiments

Distributed Buildup:

PDB ID	No. of atoms	RmsdErr	CPU time
1PTQ	402	1.781565e-012	0.912213
1HOE	558	6.620519e-011	1.974754
1LFB	641	4.849771e-012	1.660825
1PHT	811	6.675090e-012	1.769367
1POA	914	6.173696e-010	1.563789
1AX8	1003	4.161234e-008	1.793236
1F39	1534	3.480137e-012	4.303442
1RGS	2015	1.804018e-009	4.713639
1KDH	2923	5.424939e+002	18.708290

A generalized DG problem

DG problem with distance bounds

Given the lower bounds $l_{i,j}$ and upper bounds $u_{i,j}$, the problem can be formulated as:

$$\begin{aligned}
\max_{x_i, r_i} \quad & \sum_{i=1}^{n} r_i \\
\text{s.t.} \quad & \|x_i - x_j\| + r_i + r_j \le u_{i,j} \\
& \|x_i - x_j\| - r_i - r_j \ge l_{i,j} \qquad \forall (i,j) \in S \\
& r_i \ge 0, \qquad i = 1, 2, \ldots, n.
\end{aligned}$$

Atilla Sit, Zhijun Wu(2011), Solving a Generalized Distances Geometry Problem for Protein Structure Determination.

Matrix Completion for DG

Matrix Completion for DG

- Given distance matrix D, we have

$$\begin{aligned} DD = D.^2 &= (d_{i,j}^2) \\ &= (\|x_i\|^2 - 2x_i^{\mathrm{T}}x_j + \|x_j\|^2) \\ &= E + E^{\mathrm{T}} - 2XX^{\mathrm{T}}. \end{aligned}$$

where E is a rank one matrix.

Matrix Completion for DG

- Given distance matrix D, we have

$$\begin{aligned} DD = D.^2 &= (d_{i,j}^2) \\ &= (\|x_i\|^2 - 2x_i^{\mathrm{T}} x_j + \|x_j\|^2) \\ &= E + E^{\mathrm{T}} - 2XX^{\mathrm{T}}. \end{aligned}$$

  where E is a rank one matrix.

- rank(DD)=5.

Matrix Completion for DG

- Given distance matrix D, we have

$$\begin{aligned} DD = D.^2 &= (d_{i,j}^2) \\ &= (\|x_i\|^2 - 2x_i^{\mathrm{T}} x_j + \|x_j\|^2) \\ &= E + E^{\mathrm{T}} - 2XX^{\mathrm{T}}. \end{aligned}$$

  where E is a rank one matrix.

- rank(DD)=5.
- FPCA, LMaFit

A new type of MC problem

A new type of MC problem

-

$$\begin{aligned}\min_{X,x_i} \quad & \|X - \sum_{i=1}^{r} x_i x_i^{\mathrm{T}}\|_F^2 \\ \text{s.t.} \quad & X_{i,j} = M_{i,j}, \quad \forall (i,j) \in S,\end{aligned}$$

A new type of MC problem

- $$\min_{X,x_i} \quad \|X - \sum_{i=1}^{r} x_i x_i^{\mathrm{T}}\|_F^2$$
  $$\text{s.t.} \quad X_{i,j} = M_{i,j}, \quad \forall (i,j) \in S,$$
- X is symmetric.

A new type of MC problem

- $$\begin{aligned} \min_{X, x_i} \quad & \|X - \sum_{i=1}^{r} x_i x_i^{\mathrm{T}}\|_F^2 \\ \text{s.t.} \quad & X_{i,j} = M_{i,j}, \quad \forall (i,j) \in S, \end{aligned}$$

- X is symmetric.
- M has some special structure (not randomly sampled).

Conclusions and Future work

Conclusions and Future work

- What we have done:
  - propose a distributed idea and a new possible way to divide
  - propose a new model for DG
  - finish some preliminary numerical experiments

Conclusions and Future work

- What we have done:
  - propose a distributed idea and a new possible way to divide
  - propose a new model for DG
  - finish some preliminary numerical experiments
- Future work:
  - theoretical analysis on eigenvector of Laplacian matrix
  - make clear the advantage and limitation of our distributed method
  - work on the new models

Thank you for your attention!