Applications of Semidefinite Programming

Yinyu Ye

Department of Management Science and Engineering

Stanford University

Stanford, CA 94305, U.S.A.

http://www.stanford.edu/˜yyye

Max-Cut Problem

Consider the Max Cut problem on an undirected graph $G = (V, E)$ with non-negative weights w_{ij} for each edge in E (and $w_{ij} = 0$ if $(i, j) \notin E$), which is the problem of partitioning the nodes of V into two sets S and $V \setminus S$ so that

$$w(S) := \sum_{i \in S,\, j \in V \setminus S} w_{ij}$$

is maximized. A problem of this type arises from many network planning, circuit design, and scheduling applications.

Quadratic Optimization Formulation

This problem can be formulated by assigning each node a binary variable x_j:

$$z^* = \quad \text{Maximize} \quad w(\mathbf{x}) := \frac{1}{4}\sum_{i,j} w_{ij}(1 - x_i x_j)$$

(MC)

$$\text{Subject to} \quad x_j^2 = 1, \quad j = 1, ..., n.$$

The objective can be written as a quadratic function

$$w(\mathbf{x}) = \mathbf{x}^T Q \mathbf{x} = Q \bullet \mathbf{x}\mathbf{x}^T.$$

Randomized Rounding: The Coin-Toss Method

Generate a random vector $\mathbf{u} \in N(0, I)$, and let

$$\hat{\mathbf{x}} = \text{sign}(\mathbf{u}),$$

where

$$\text{sign}(x_j) = \begin{cases} 1 & \text{if } x_j \geq 0 \\ -1 & \text{otherwise.} \end{cases}$$

This is equivalent to that each node be selected to one side, or $\hat{x}_j = 1$, independently with probability $.5$.

$$\text{E}[w(\hat{\mathbf{x}})] \geq 0.5 \cdot z^*.$$

Semi-definite relaxation

Let $X = \mathbf{x}\mathbf{x}^T$. Then the QP can be relaxed to

$$
\begin{array}{rll}
z^{SDP} := & \text{Maximize} & Q \bullet X \\
& \text{s.t.} & I_j \bullet X = 1, \; j = 1, ..., n, \\
& & X \succeq 0, \; (\text{Rank}(X) = 1).
\end{array} \tag{1}
$$

The dual is

$$
\begin{array}{rll}
z^{SDP} = & \text{Minimize} & \mathbf{e}^T \mathbf{y} \\
& \text{s.t.} & D(\mathbf{y}) \succeq Q.
\end{array} \tag{2}
$$

SDP Based Randomized Rounding

Let $V = (\mathbf{v}_1, \ldots, \mathbf{v}_n) \in \mathcal{R}^{n\times n}$, i.e., $\mathbf{v}_j$ is the jth column of V, such that $X^* = V^T V$, where X^* is a maximizer of the SDP relaxation problem.

Generate a random vector $\mathbf{u} \in N(0, I)$:

$$\hat{\mathbf{x}} = \text{sign}(V^T \mathbf{u}),$$

Approximation analysis

Then, one can prove from Sheppard (1900):

$$\mathsf{E}[\hat{x}_i \hat{x}_j] = \frac{2}{\pi} \arcsin(X^*_{ij}), \quad i, j = 1, 2, \ldots, n.$$

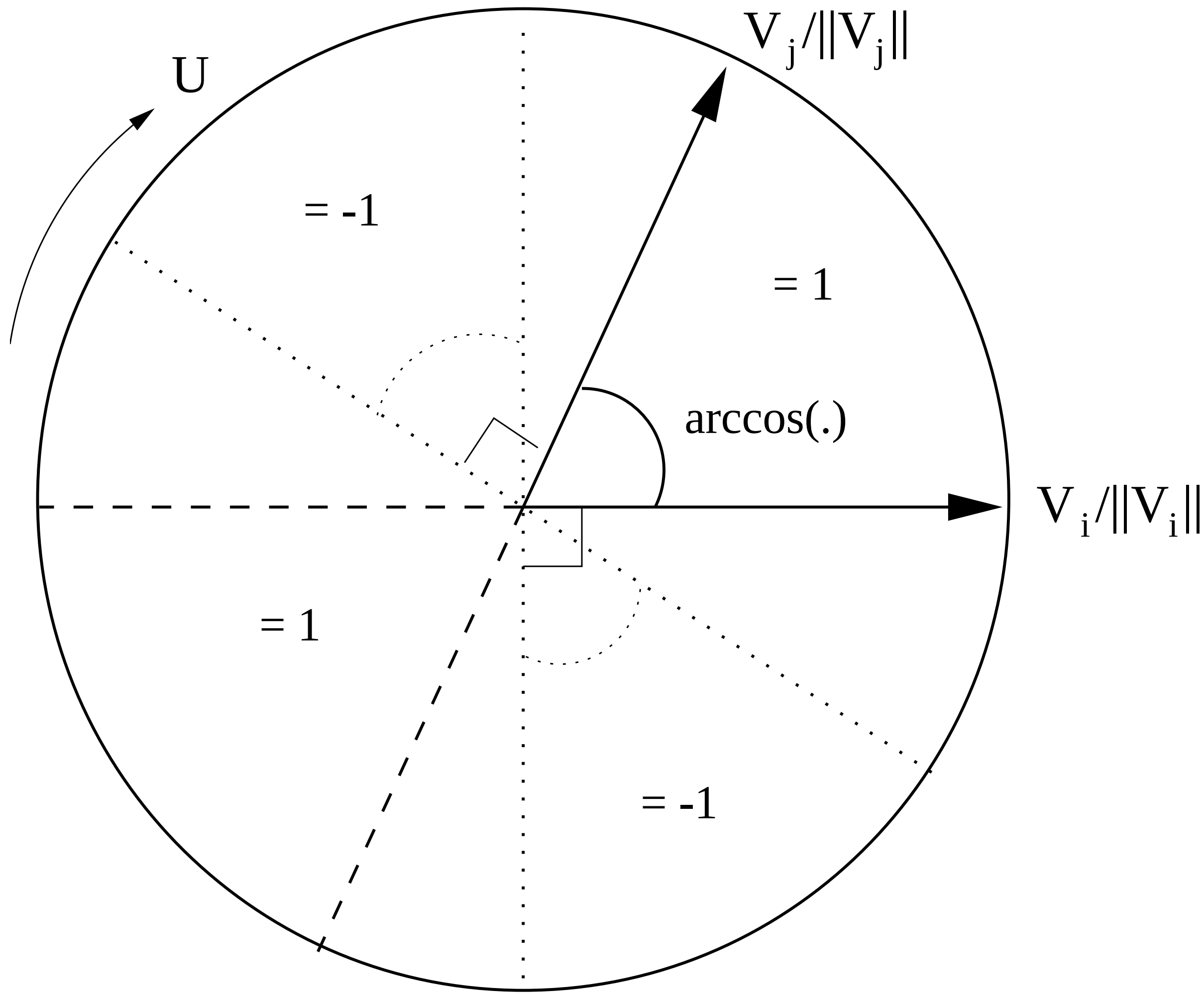


Figure 1: Illustration of the product $\sigma(\frac{\mathbf{v}_i^T\mathbf{u}}{\|\mathbf{v}_i\|}) \cdot \sigma(\frac{\mathbf{v}_j^T\mathbf{u}}{\|\mathbf{v}_j\|})$ on the 2-dimensional unit circle, where $\mathbf{u}$ is uniformly generated along the circle.

Analyses

Lemma 1 *For* $x \in [-1, 1)$

$$\frac{1 - (2/\pi) \cdot \arcsin(x)}{1 - x} \geq .878.$$

Lemma 2 *Let* $X \succeq 0$ *and* $d(X) \leq 1$*. Then* $\arcsin[X] \succeq X$*.*

Final Results

Theorem 1 *We have*

i) *If Q is a Laplacian matrix, then*

$$\mathsf{E}(\hat{\mathbf{x}}^T Q \hat{\mathbf{x}}) \geq .878 z^{SDP} \geq .878 z^*,$$

so that

$$z^* \geq .878 z^{SDP}.$$

ii) *If Q is positive semidefinite*

$$\mathsf{E}(\hat{\mathbf{x}}^T Q \hat{\mathbf{x}}) \geq \frac{2}{\pi} z^{SDP} \geq \frac{2}{\pi} z^*,$$

so that

$$z^* \geq \frac{2}{\pi} z^{SDP}.$$

Generalized QP SDP Relaxation

$$
\begin{array}{rll}
z^{SDP} := & \text{Maximize} & Q \bullet X \\
& \text{s.t.} & I_j \bullet X \{= \text{ or } \leq\} 1, \ \forall j, \\
& & X \succeq 0.
\end{array}
$$

$$
\mathsf{E}(\hat{\mathbf{x}}^T Q \hat{\mathbf{x}}) \geq \frac{2}{\pi} z^{SDP} \geq \frac{2}{\pi} z^*.
$$

Graph Realization

Given a graph $G = (V, E)$ and sets of non–negative weights, say $\{d_{ij} : (i, j) \in E\}$ and $\{\theta_{ilj} : (i, l, j) \in \Theta\}$, the goal is to compute a realization of G in the Euclidean space $\mathbf{R}^d$ for a given low dimension d, i.e.

- to place the vertices of G in $\mathbf{R}^d$ such that
- the Euclidean distance between every pair of adjacent vertices (i, j) equals (or bounded) by the prescribed weight $d_{ij} \in E$, and
- the angle between edges (i, l) and (j, l) equals (or bounded) by the prescribed weight $\theta_{ilj} \in \Theta$.

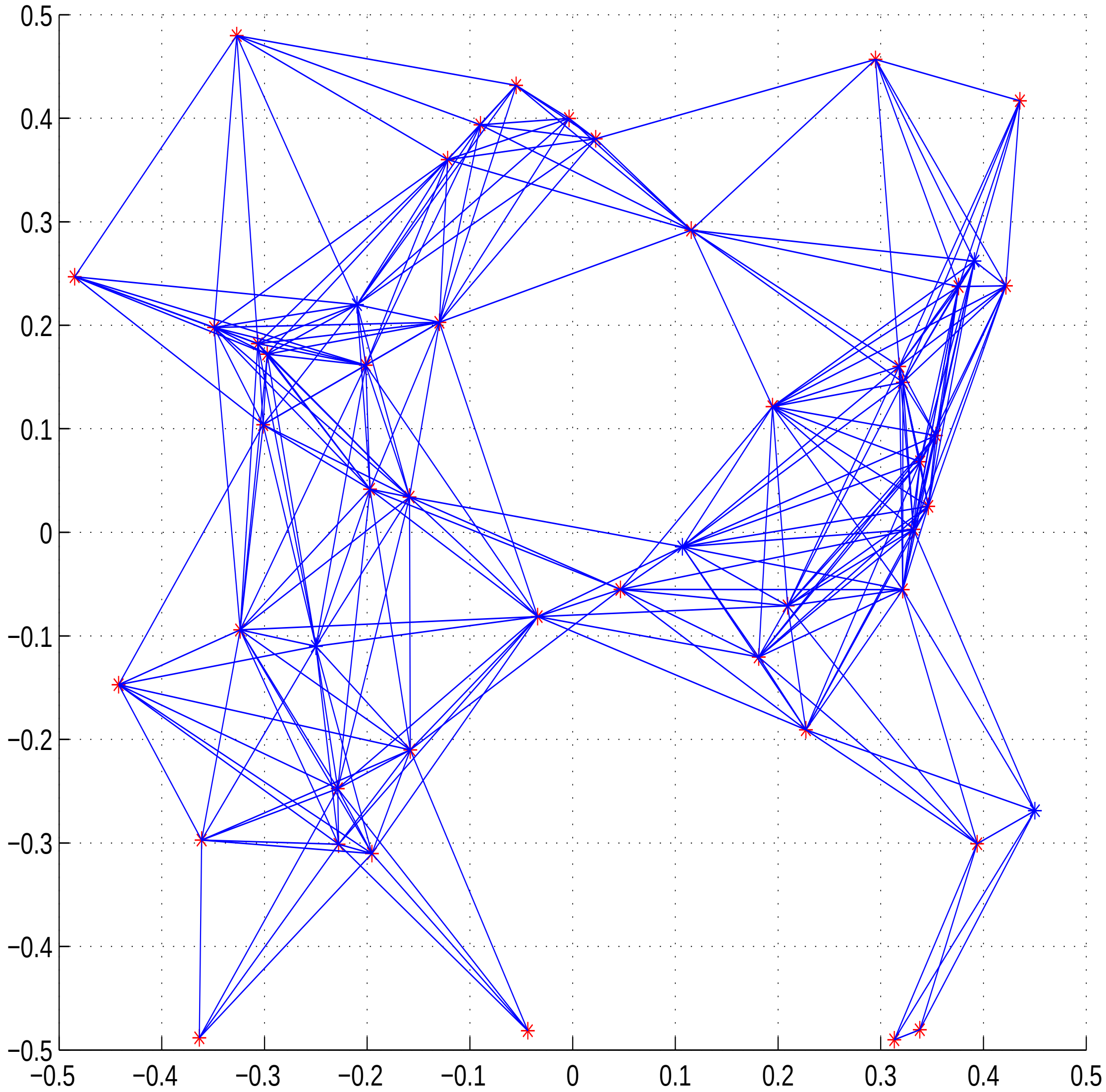


Figure 2: 50-node 2-D Sensor Localization

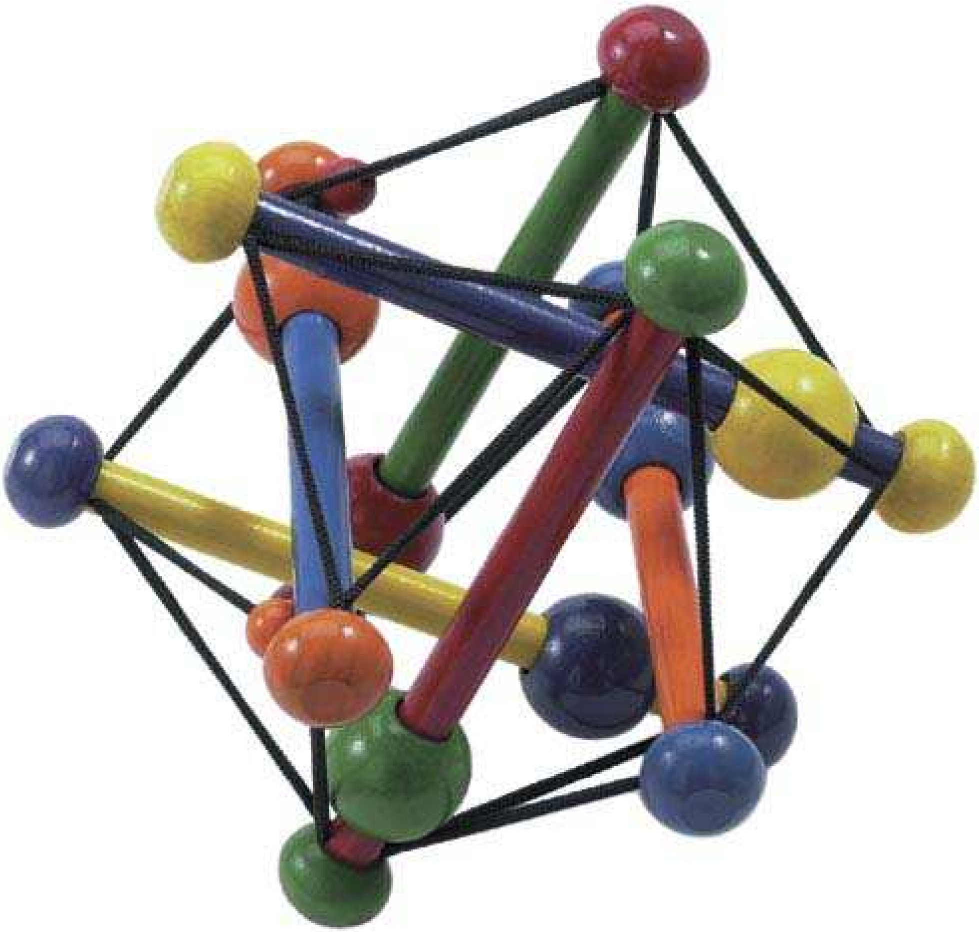


Figure 3: A 3-D Tensegrity Graph Realization; picture provided by Anstreicher

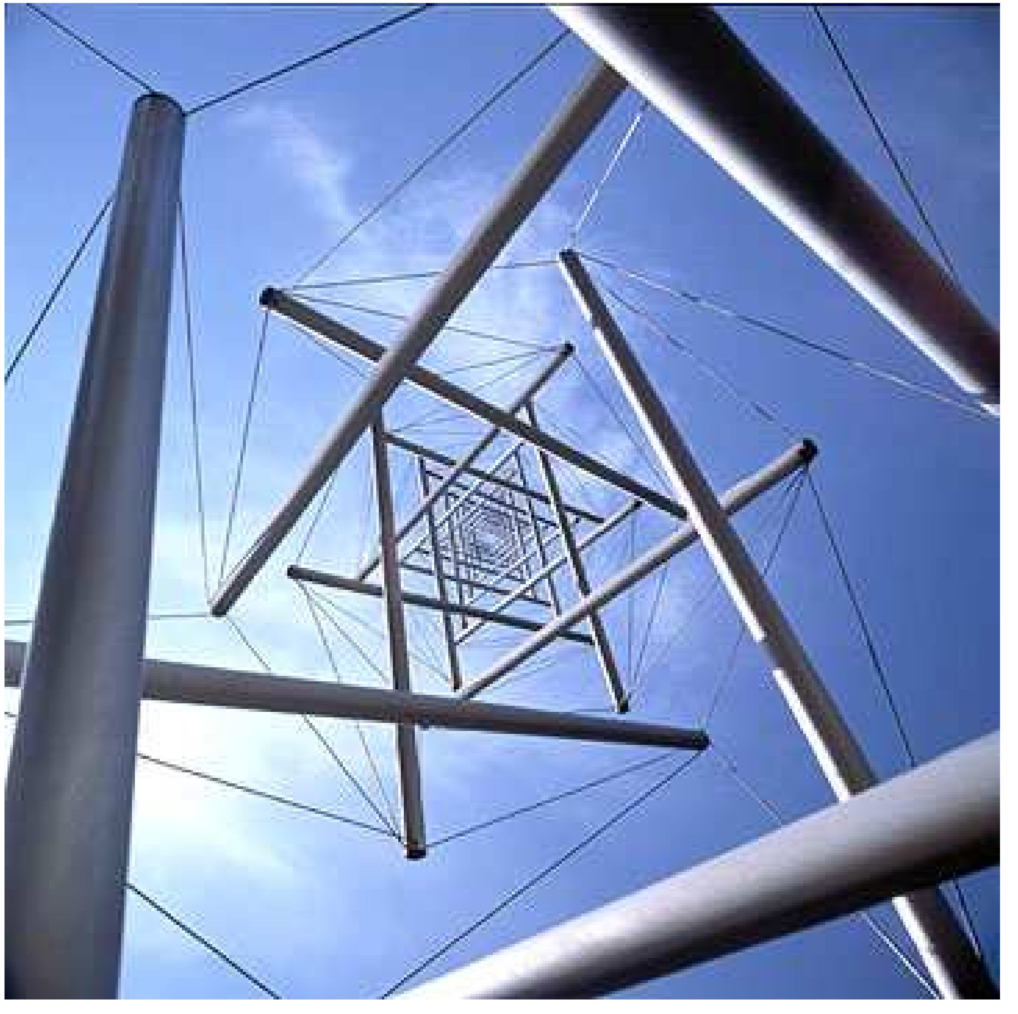


Figure 4: Tensegrity Graph: A Needle Tower; picture provided by Anstreicher

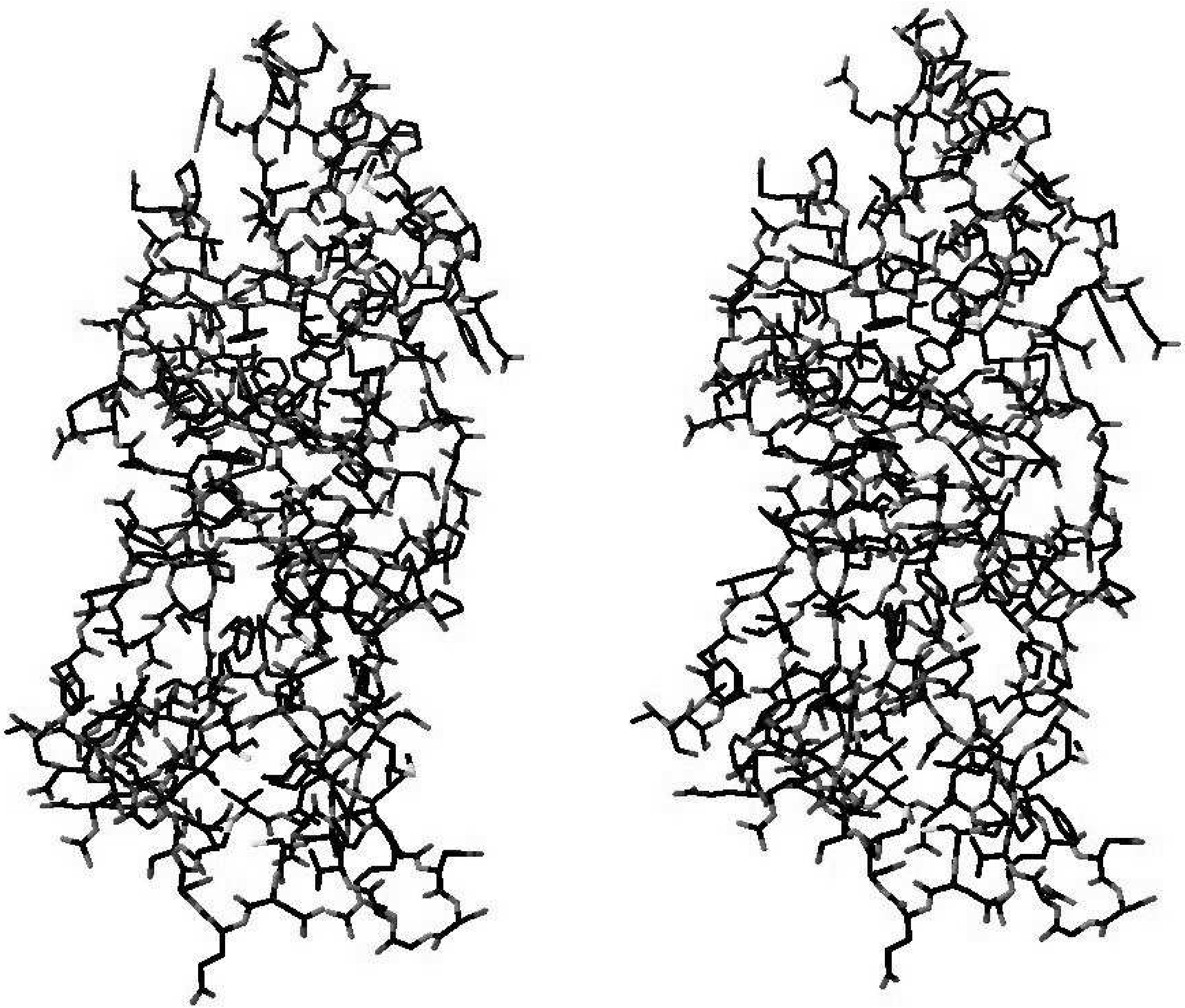


Figure 5: Molecular Conformation: 1F39(1534 atoms) with 85% of distances below $6\mathbf{r}$A and 10% noise on upper and lower bounds

Quadratically-Constrained Problems

Given $d_{ij} \in N_x$ and $v_{ilj} \in \Theta$, find $\mathbf{x}_i \in \mathbf{R}^d$ such that

$$
\begin{array}{lll}
\|\mathbf{x}_i - \mathbf{x}_j\|^2 & (\leq) = (\geq) & d_{ij}^2, \; \forall\, (i,j) \in N_x, \; i < j, \\
(\mathbf{x}_i - \mathbf{x}_l)^T(\mathbf{x}_j - \mathbf{x}_l) & (\leq) = (\geq) & v_{ilj}, \; \forall\, (i,l,j) \in \Theta.
\end{array}
$$

A Simplified Sensor Localization Model

Given $\mathbf{a}_k \in \mathbf{R}^d$, $d_{ij} \in N_x$, and $\hat{d}_{kj} \in N_a$, find $\mathbf{x}_i \in \mathbf{R}^d$ such that

$$\begin{aligned} \|\mathbf{x}_i - \mathbf{x}_j\|^2 &= d_{ij}^2, \ \forall\,(i,j) \in N_x,\ i < j, \\ \|\mathbf{a}_k - \mathbf{x}_j\|^2 &= \hat{d}_{kj}^2, \ \forall\,(k,j) \in N_a, \end{aligned}$$

(ij) ((kj)) connects points $\mathbf{x}_i$ and $\mathbf{x}_j$ ($\mathbf{a}_k$ and $\mathbf{x}_j$) with an edge whose Euclidean length is d_{ij} ($\hat{d}_{kj}$).

Does the system have a localization or realization of all $\mathbf{x}_j$'s? Is the localization unique? Is there a certification for the solution to make it reliable or trustworthy? Is the system partially localizable with certification?

Global and Nonlinear Least Squares

$$\min \quad \sum_{i,j\in N_x} (\|\mathbf{x}_i - \mathbf{x}_j\|^2 - d_{ij}^2)^2 + \sum_{k,j\in N_a} (\|\mathbf{a}_k - \mathbf{x}_j\|^2 - \hat{d}_{kj}^2)^2$$

$$\min \quad \sum_{i,j\in N_x} (\|\mathbf{x}_i - \mathbf{x}_j\| - d_{ij})^2 + \sum_{k,j\in N_a} (\|\mathbf{a}_k - \mathbf{x}_j\| - \hat{d}_{kj})^2$$

Matrix Representation

Let $X = [\mathbf{x}_1 \ \mathbf{x}_2 \ \ldots \ \mathbf{x}_n]$ be the $2 \times n$ matrix that needs to be determined. Then

$$\|\mathbf{x}_i - \mathbf{x}_j\|^2 = (\mathbf{e}_i - \mathbf{e}_j)^T X^T X (\mathbf{e}_i - \mathbf{e}_j) \text{ and}$$

$$\|\mathbf{a}_k - \mathbf{x}_j\|^2 = (\mathbf{a}_k; -\mathbf{e}_j)^T [I \ \ X]^T [I \ \ X] (\mathbf{a}_k; -\mathbf{e}_j),$$

where $\mathbf{e}_j$ is the vector of all zero except 1 at the jth position.

$$\begin{aligned}
&(\mathbf{e}_i - \mathbf{e}_j)^T Y (\mathbf{e}_i - \mathbf{e}_j) = d_{ij}^2, \ \forall\, i, j \in N_x, \ i < j, \\
&(\mathbf{a}_k; -\mathbf{e}_j)^T \begin{pmatrix} I & X \\ X^T & Y \end{pmatrix} (\mathbf{a}_k; -\mathbf{e}_j) = \hat{d}_{kj}^2, \ \forall\, k, j \in N_a, \\
&Y = X^T X.
\end{aligned}$$

SDP Relaxation

Change

$$Y = X^T X$$

to

$$Y \succeq X^T X.$$

This matrix inequality is equivalent to

$$\begin{pmatrix} I & X \\ X^T & Y \end{pmatrix} \succeq 0.$$

This matrix has rank at least 2; if it's 2, then $Y = X^T X$, and the converse is also true.

SDP Standard Form

$$Z = \begin{pmatrix} I & X \\ X^T & Y \end{pmatrix}.$$

Find a symmetric matrix $Z \in \mathbf{R}^{(2+n)\times(2+n)}$ such that

$$\begin{aligned}
& Z_{1:2,1:2} = I \\
& (\mathbf{0}; (\mathbf{e}_i - \mathbf{e}_j))(\mathbf{0}; (\mathbf{e}_i - \mathbf{e}_j))^T \bullet Z = d_{ij}^2, \ \forall\, i, j \in N_x, \ i < j, \\
& (\mathbf{a}_k; -\mathbf{e}_j)(\mathbf{a}_k; -\mathbf{e}_j)^T \bullet Z = \hat{d}_{kj}^2, \ \forall\, k, j \in N_a, \\
& Z \succeq 0.
\end{aligned}$$

If every sensor point is connected, directly or indirectly, to an anchor point, then the solution set must be bounded.

Unique Localizability

A sensor network is 2-uniquely-localizable if there is a unique localization in $\mathbf{R}^2$ and there is no $\mathbf{x}_j \in \mathbf{R}^h,\ j = 1, ..., n$, where $h > 2$, such that

$$\|\mathbf{x}_i - \mathbf{x}_j\|^2 = d_{ij}^2,\ \forall\, i, j \in N_x,\ i < j,$$
$$\|(\mathbf{a}_k; \mathbf{0}) - \mathbf{x}_j\|^2 = \hat{d}_{kj}^2,\ \forall\, k, j \in N_a.$$

The latter says that the problem cannot be localized in a higher dimension space where anchor points are simply augmented to $(\mathbf{a}_k; \mathbf{0}) \in \mathbf{R}^h$, $k = 1, ..., m$.

Figure 6: One sensor-Two anchors: Not localizable

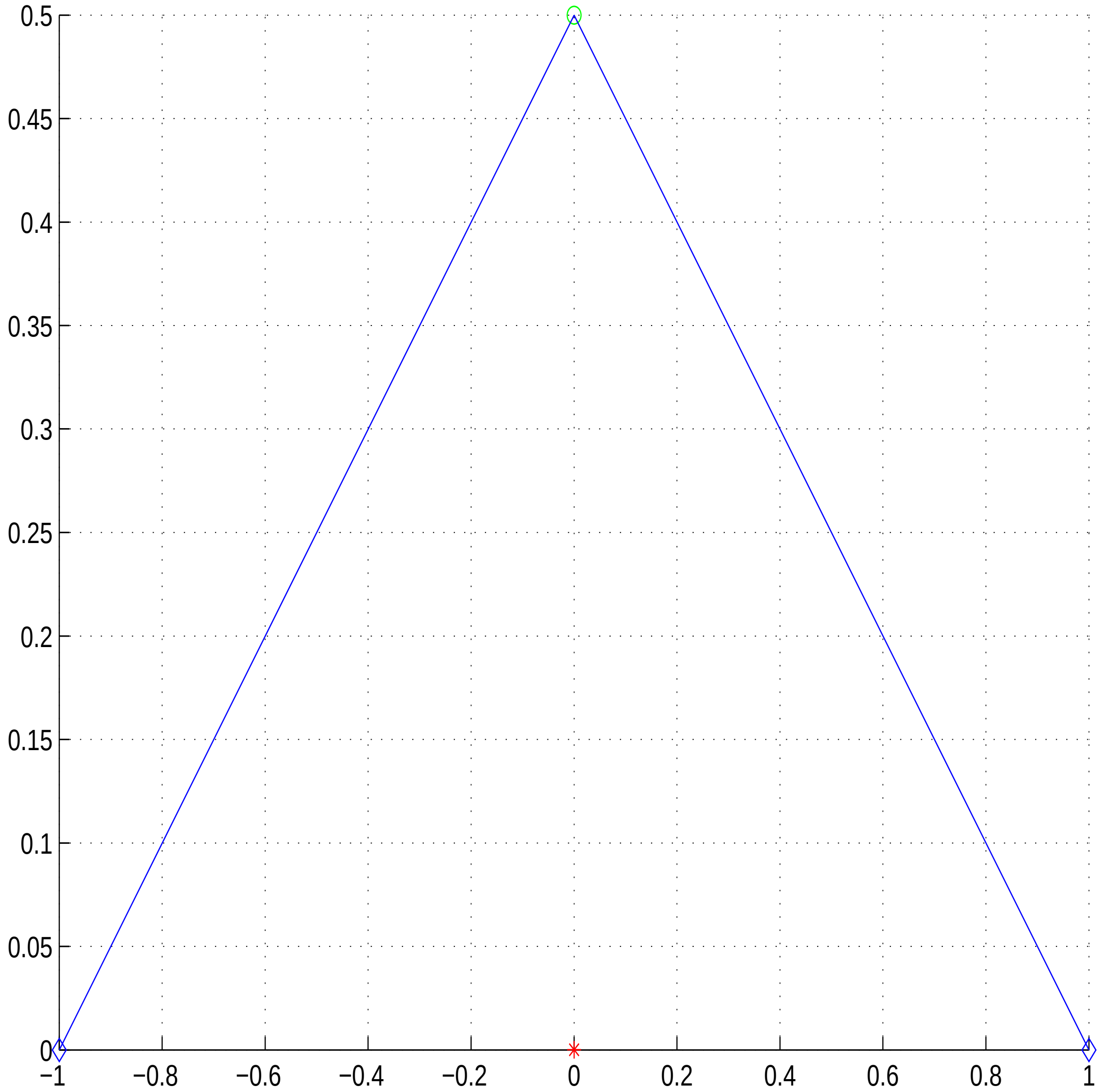


Figure 7: Two sensor-Three anchors: Strongly Localizable

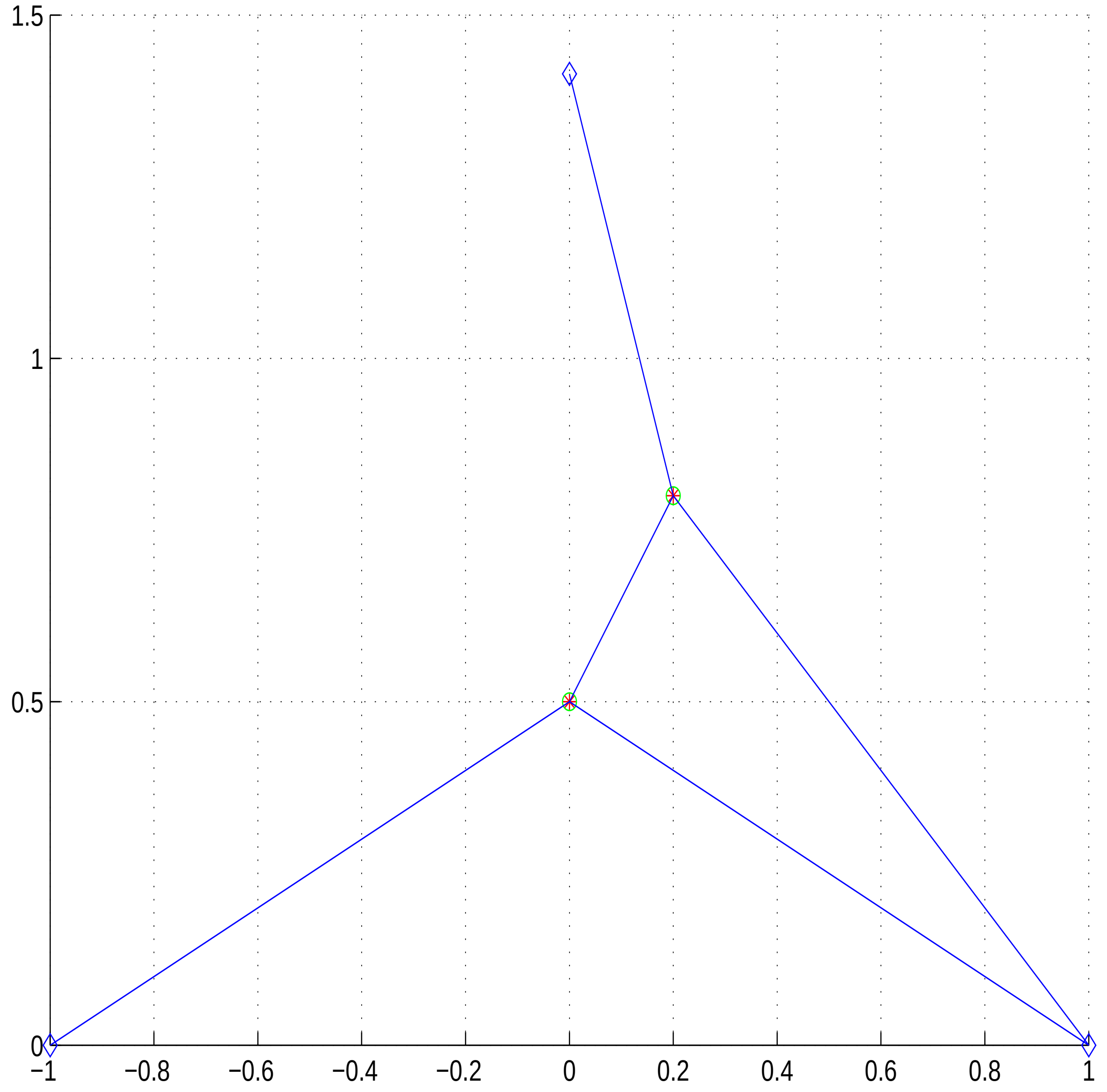


Figure 8: Two sensor-Three anchors: Localizable but not Strongly

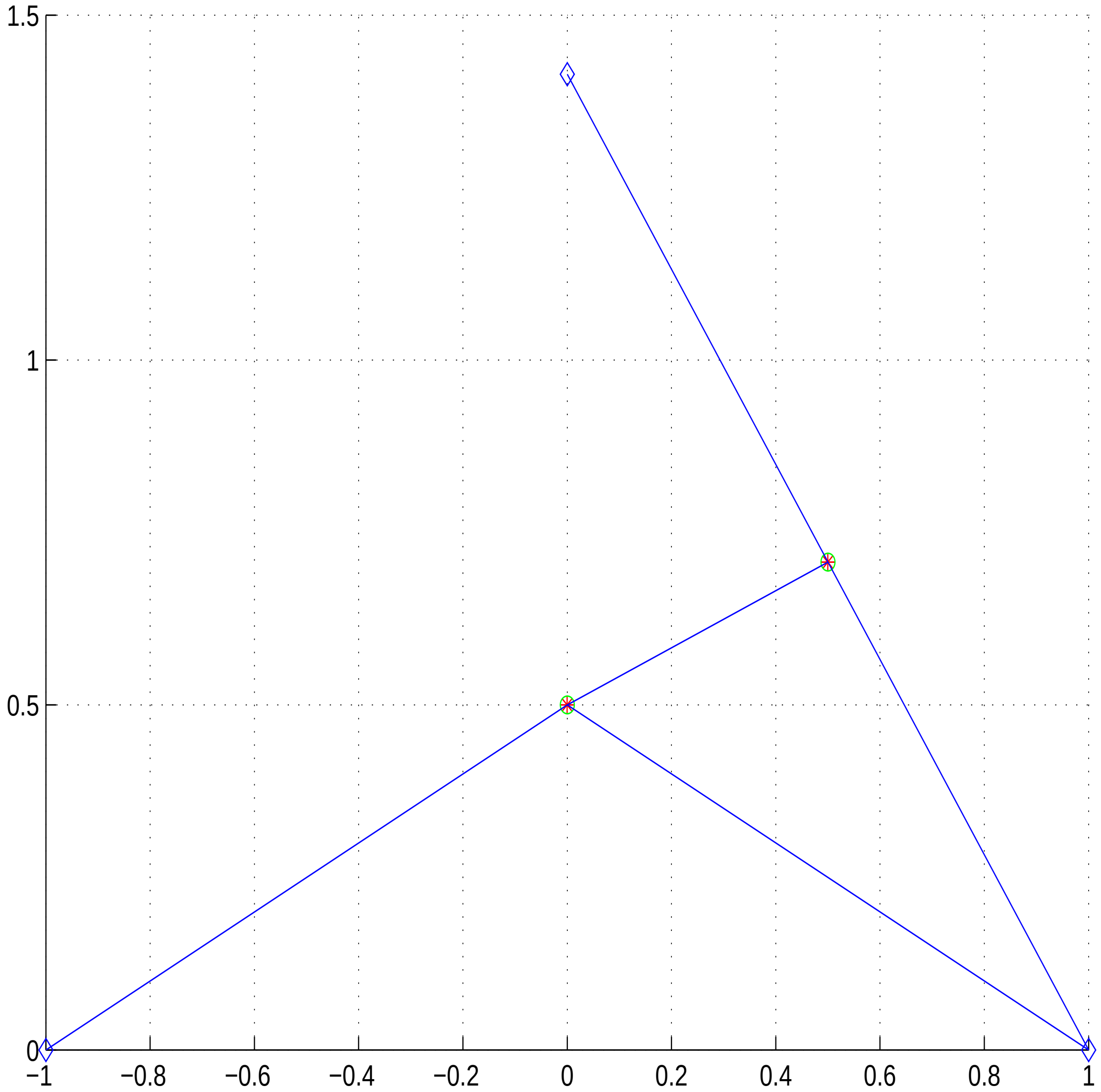


Figure 9: Two sensor-Three anchors: Not localizable

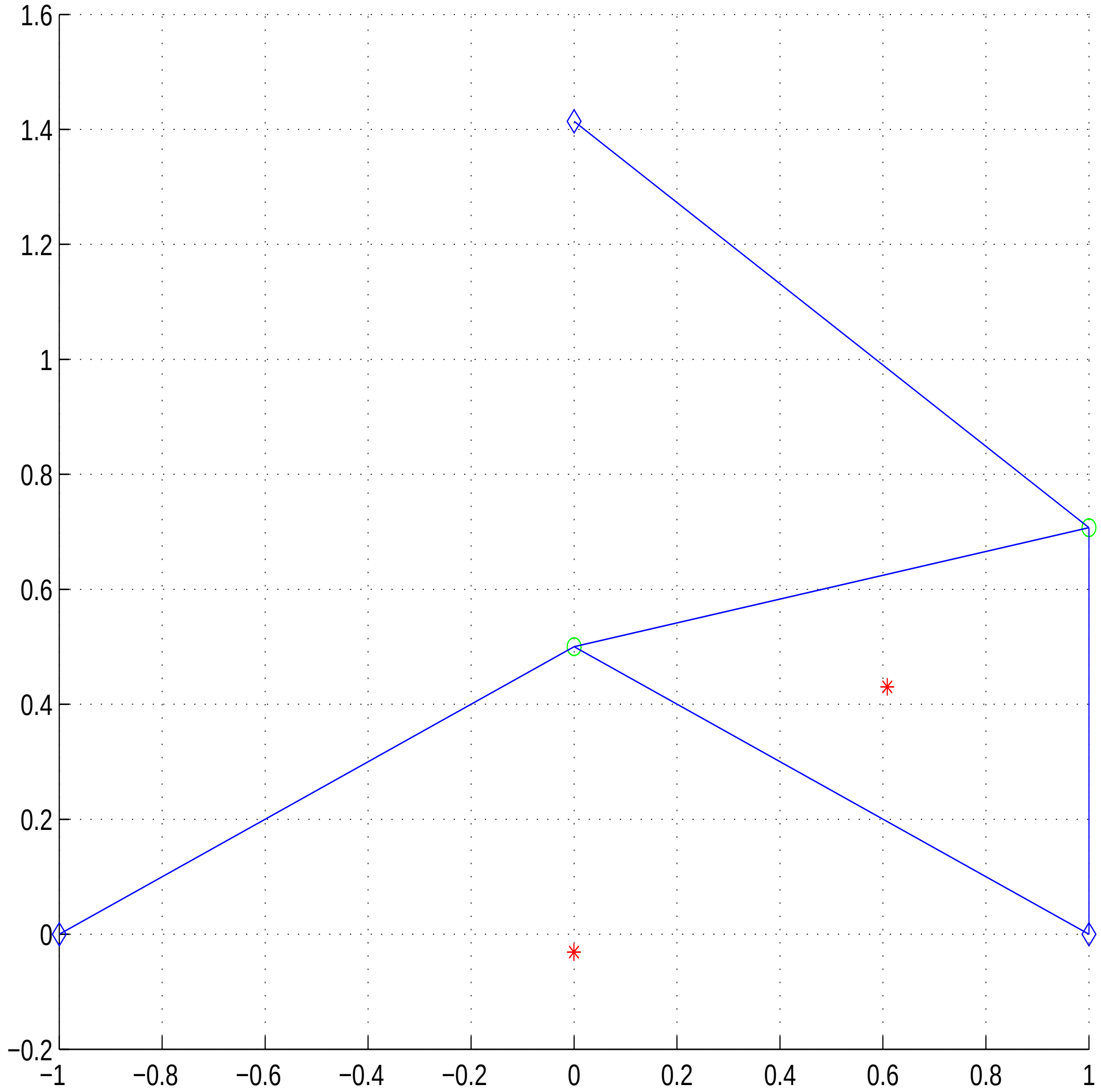
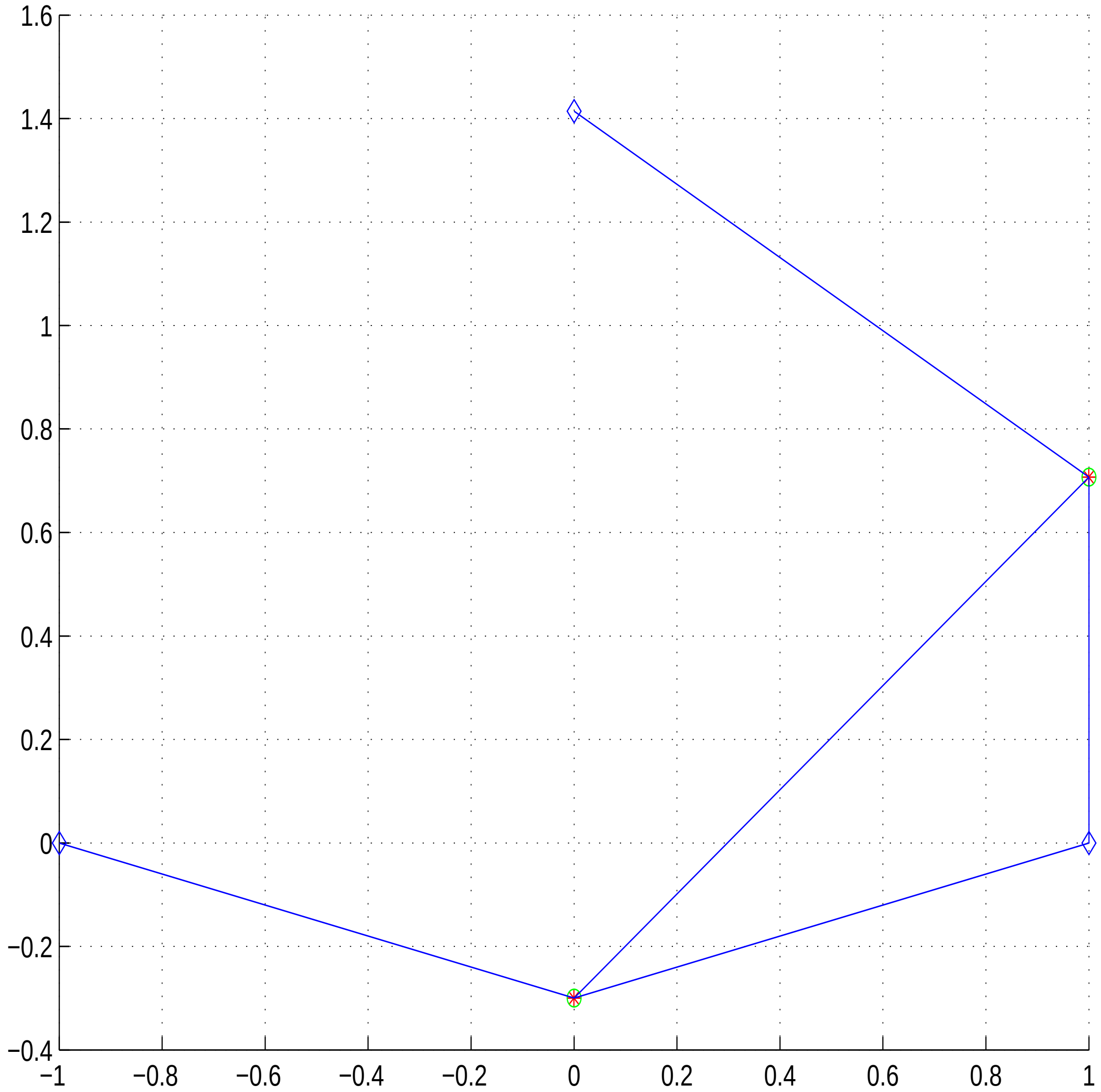


Figure 10: Two sensor-Three anchors: Strongly Localizable



Uniquely-Localizable Problems

Theorem 2

- *If every edge length is specified, then the sensor network is 2-uniquely-localizable (Schoenberg 1942).*
- *There is a sensor network, with $O(n)$ edge lengths specified, that is 2-uniquely-localizable.*
- *If one sensor with its edge lengths to at least three anchors (in general positions) specified, then it is 2-uniquely-localizable.*

One sensor and three anchors

Find $\mathbf{x}_1 \in \mathbf{R}^2$ such that

$$\|\mathbf{a}_k - \mathbf{x}_1\|^2 = d_{kj}^2, \text{ for } k = 1, 2, 3,$$

Let $\bar{\mathbf{x}}_1$ be the true position of $\mathbf{x}_1$.

SDP Standard Form

$$
\begin{aligned}
&(1;0;0)(1;0;0)^T \bullet Z = 1,\\
&(0;1;0)(0;1;0)^T \bullet Z = 1,\\
&(1;1;0)(1;1;0)^T \bullet Z = 2,\\
&(\mathbf{a}_k;-1)(\mathbf{a}_k;-1)^T \bullet Z = d_{k1}^2, \text{ for } k = 1,2,3,\\
&Z \succeq 0.
\end{aligned}
$$

$$
\bar{Z} = \begin{pmatrix} I & \bar{\mathbf{x}}_1 \\ \bar{\mathbf{x}}_1^T & \bar{\mathbf{x}}_1^T \bar{\mathbf{x}}_1 \end{pmatrix} = (I,\ \bar{\mathbf{x}}_1)^T (I,\ \bar{\mathbf{x}}_1)
$$

is a feasible solution for the relaxation.

The Dual Slack Matrix

$$\begin{pmatrix} \begin{pmatrix} w_1 + w_3 & w_3 \\ w_3 & w_2 + w_3 \end{pmatrix} + \sum_{k=1}^{3} w_{k1} a_k a_k^T & -\sum_{k=1}^{3} w_{k1} a_k \\ -(\sum_{k=1}^{3} w_{k1} a_k)^T & w_{11} + w_{21} + w_{31} \end{pmatrix} \succeq 0.$$

Does an optimal matrix U have rank 1 with

$$w_1 + w_2 + w_3 + \sum_{k=1}^{3} w_{k1} d_{k1}^2 = 0$$

An Optimal Dual Slack Matrix

If we choose $w_\bullet$'s such that

$$\bar{U} = (-\bar{\mathbf{x}}_1; 1)(-\bar{\mathbf{x}}_1; 1)^T,$$

then, $\bar{U} \succeq 0$ and $\bar{U} \bullet \bar{X} = 0$ so that $\bar{U}$ is an optimal slack matrix for the dual and its rank is 1.

How to Select w's

Let

$$\sum_{k=1}^3 w_{k1} a_k = \bar{\mathbf{x}}_1 \qquad \text{or} \qquad \sum_{k=1}^3 w_{k1}(\mathbf{a}_k - \bar{\mathbf{x}}_1) = 0$$

$$w_{11} + w_{21} + w_{31} = 1. \qquad\qquad w_{11} + w_{21} + w_{31} = 1.$$

This system always has a solution if $\mathbf{a}_k$ is not co-linear.

Then, select

$$\begin{pmatrix} w_1 + w_3 & w_3 \\ w_3 & w_2 + w_3 \end{pmatrix} = \bar{\mathbf{x}}_1 \bar{\mathbf{x}}_1^T - \sum_{k=1}^3 w_{k1} \mathbf{a}_k \mathbf{a}_k^T$$

Other Conditions?

Even if a_k is co-linear, the system

$$\sum_{k=1}^{3} w_{k1}(\mathbf{a}_k - \bar{\mathbf{x}}_1) = 0$$

$$w_{11} + w_{21} + w_{31} = 1$$

has a solution $w_\bullet$ if $\bar{\mathbf{x}}_1$ on the same line.

Physical interpretation: w_{ij} is a force on the edge and all forces are balanced.

The objective represents the work of the system.

ULPs can be localized in polynomial time

Theorem 3 *The following statements are equivalent:*

1. *The sensor network is 2-uniquely-localizable;*
2. *The max-rank solution of the SDP relaxation has rank 2;*
3. *The solution matrix has $Y = X^T X$ or Trace$(Y - X^T X) = 0$.*

When an optimal dual (stress) slack matrix has rank n, then the problem is 2-strongly-localizable.

If one sensor with its edge lengths to at least three anchors (in general positions) specified, then it is 2-strongly-localizable

Proof of Theorem

We need to prove that if the primal feasible matrix generated from the interior-point algorithm has rank 2, that is, $\bar{Y} = \bar{X}^T \bar{X}$, then the solution for the original problem must be unique.

First, every feasible matrix has rank at least 2 since $Y \succeq X^T X$. Second, since the matrix solution computed from the interior-point algorithm has the maximal rank and it is 2, we conclude that every feasible matrix has rank exactly 2.

Suppose that the system has two rank-2 feasible matrices:

$$Z_1 = \begin{pmatrix} I & X_1 \\ X_1^T & X_1^T X_1 \end{pmatrix} \quad \text{and} \quad Z_2 = \begin{pmatrix} I & X_2 \\ X_2^T & X_2^T X_2 \end{pmatrix}$$

Consider $Z = \alpha Z_1 + \beta Z_2$, where $\alpha + \beta = 1$ and $\alpha, \beta > 0$. Then Z is a feasible solution and its rank must be 2.

Proof continued

$$Z = \begin{pmatrix} I & \alpha X_1 + \beta X_2 \\ \alpha X_1^T + \beta X_2^T & \alpha X_1^T X_1 + \beta X_2^T X_2 \end{pmatrix} =$$

$$\begin{pmatrix} I & \alpha X_1 + \beta X_2 \\ \alpha X_1^T + \beta X_2^T & (\alpha X_1 + \beta X_2)^T (\alpha X_1 + \beta X_2) \end{pmatrix}$$

Thus,

$$0 = \alpha X_1^T X_1 + \beta X_2^T X_2 - (\alpha X_1 + \beta X_2)^T (\alpha X_1 + \beta X_2) =$$

$$\alpha\beta (X_1 - X_2)^T (X_1 - X_2)$$

or

$$\|X_1 - X_2\| = 0.$$

Localize All Localizable Points

Theorem 4 *If a problem (graph) contains a subproblem (subgraph) that is localizable, then the submatrix solution corresponding to the subproblem in the SDP solution has rank 2. That is, the SDP relaxation computes a solution that localize all possibly localizable unknown sensor points.*

Implication: Diagonals of "co-variance" matrix

$$Y^* - (X^*)^T X^*,$$

$Y^*_{jj} - \|\mathbf{x}^*_j\|^2$, can be used as a measure to see whether jth sensor's estimated position is reliable or not.

Uncertainty Analysis and Confidence Measure

Alternatively, each $\mathbf{x}_j$'s can be viewed as uncertain points from the incomplete distance measures. Then the solution to the SDP problem provides the first and second moment estimation.

Generally, $\mathbf{x}_j^*$ is a point estimate of $\mathbf{x}_j$ and Y_{ij}^* is a point estimate $\mathbf{x}_i^T\mathbf{x}_j$.

Consequently,

$$Y_{jj}^* - \|\mathbf{x}_j^*\|^2,$$

which is the individual variance estimation of sensor j, gives an interval estimation for its true position.

SDP Relaxation of Least Squares

$$
\begin{array}{ll}
\text{minimize} & \sum_{(i,j)\in N_x} \alpha_{ij}^2 + \sum_{k,j\in N_a} \alpha_{kj}^2 \\
\text{subject to} & \|\mathbf{x}_i - \mathbf{x}_j\|^2 = d_{ij}^2 + \alpha_{ij}, \text{ for } (i,j) \in N_x, \\
& \|\mathbf{a}_k - \mathbf{x}_j\|^2 = d_{kj}^2 + \alpha_{kj}, \text{ for } (k,j) \in N_a.
\end{array}
$$

$$
\begin{array}{ll}
\text{minimize} & \sum_{(i,j)\in N_x} \alpha_{ij}^2 + \sum_{(k,j)\in N_a} \alpha_{kj}^2 \\
\text{subject to} & Z_{1:2,1:2} = I, \\
& (\mathbf{0}; \mathbf{e}_i - \mathbf{e}_j)(\mathbf{0}; \mathbf{e}_i - \mathbf{e}_j)^T \bullet Z + \alpha_{ij} = d_{ij}^2, \ \forall\, (i,j) \in N_x, \\
& (\mathbf{a}_k; -\mathbf{e}_j)(\mathbf{a}_k; -\mathbf{e}_j)^T \bullet Z + \alpha_{kj} = d_{kj}^2, \ \forall\, (k,j) \in N_a, \\
& Z \succeq 0.
\end{array}
$$

Matlab SDP Computation

SDP solvers used were SeDuMi (Sturm, 2001) and DSDP2.0 (Benson et al. 1998).

In the computational experiments:

$$d_{ij} = \text{true}d_{ij} \cdot (1 + randn(1) \cdot nf)$$

We construct SDP models to minimize max-likelihood errors when distance measures are noisy (Biswas et al 2005, Biswas 2007).

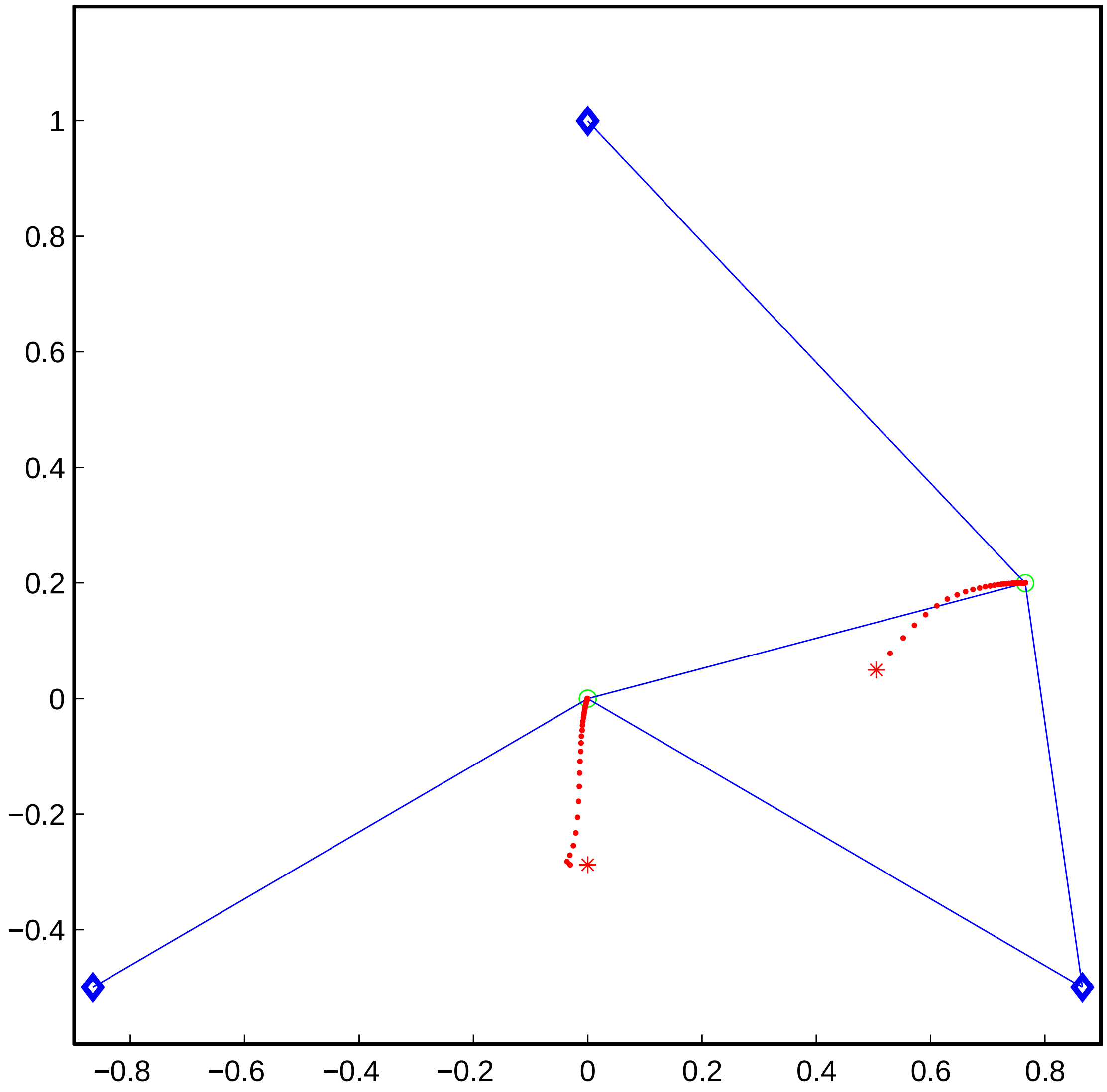
Rounding the SDP solution

- When measurement noises exist, the SDP solution almost always has a high rank. How to round the high-rank solution into a low rank?

- Gradient-based local search: using the SDP relaxation solution as the initial point, apply the steepest descent method to

$$\sum_{i,j\in N_x}(\|\mathbf{x}_i-\mathbf{x}_j\|^2-d_{ij}^2)^2+\sum_{k,j\in N_a}(\|\mathbf{a}_k-\mathbf{x}_j\|^2-d_{kj}^2)^2$$

Figure 11: Gradient search trajectories: Two sensor-Three anchor Example



NSDP Decomposition: Further Relaxation

Replace

$$(C1): \quad \begin{pmatrix} I & X \\ X^T & Y \end{pmatrix} \succeq 0 \text{ by}$$

$$(C2): \quad \begin{pmatrix} I & \mathbf{x}_i & \mathbf{x}_{i_1} & \cdots & \mathbf{x}_{i_{d(i)}} \\ \mathbf{x}_i^T & Y_{ii} & Y_{ii_1} & \cdots & Y_{ii_{d(i)}} \\ \mathbf{x}_{i_1}^T & Y_{i_1 i} & Y_{i_1 i_1} & \cdots & Y_{i_1 i_{d(i)}} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \mathbf{x}_{i_{d(i)}}^T & Y_{i_{d(i)} i} & Y_{i_{d(i)} i_1} & & Y_{i_{d(i)} i_{d(i)}} \end{pmatrix} \succeq 0, \quad \forall i,$$

where $(i, i_{ik}) \in N_x$ and $d(i)$ is the degree of sensor node i.

ESDP Decomposition: Further Relaxation

Replace

$$(C1): \quad \begin{pmatrix} I & X \\ X^T & Y \end{pmatrix} \succeq 0;$$

by

$$(C2): \quad \begin{pmatrix} I & \mathbf{x}_i & \mathbf{x}_j \\ \mathbf{x}_i^T & Y_{ii} & Y_{ij} \\ \mathbf{x}_j^T & Y_{ji} & Y_{jj} \end{pmatrix} \succeq 0, \quad \forall (i,j) \in N_x.$$

Relaxation Analyses

A undirected graph is a chordal graph if every cycle of length greater than three has a chord.

A square matrix is called to be partial symmetric if it is symmetric to the extent of its specified entries, i.e., if the (i, j) entry of the matrix is specified, then so is the (j, i) entry and the two are equal. A partial semi-definite matrix is a partial symmetric matrix and every fully specified principal submatrix is positive semi-definite.

Lemma 3 *(Hogben 2001) Every partial positive semi-definite matrix with undirected graph G has positive semi-definite completion if and only if G is chordal.*

The Equivalence Theorem for NSDP

Theorem 5 *(Wang, Zheng, Boyd and Y [2006]) Suppose the undirected graph of sensor nodes with edge set N_x is chordal, then SDP and NSDP relaxations are equivalent.*

The Trace Theorem for ESDP

Theorem 6 *Let*

$$Z = \begin{pmatrix} I & X \\ X^T & Y \end{pmatrix}$$

be a solution of ESDP computed by a path-following method. If the diagonal entry or individual trace

$$(Y - X^T X)_{\bar{i}\bar{i}} = 0$$

then the $\bar{i}$th column of X, $\mathbf{x}_{\bar{i}}$, must be the true location of the $\bar{i}$th sensor, and $\mathbf{x}_{\bar{i}}$ is invariant over all solutions Z for ESDP.

Matlab ESDP Computation

SDP solvers used were SeDuMi (Sturm, 2001) and DSDP2.0 (Benson et al. 1998).

In the computational experiments:

$$d_{ij} = \text{true}d_{ij} \cdot (1 + randn(1) \cdot nf)$$

More on Rounding the SDP solution

- When measurement noises exist, the SDP solution almost always has a high rank. How to round the high-rank solution into a low rank?
- Add a suitable objective

$$
\begin{array}{lll}
(SDP) & \text{min} & C \bullet Z \\
 & \text{subject to} & A_i \bullet Z = b_i, i = 1, 2, ..., m, \;\; Z \succeq 0.
\end{array}
$$

More Applications: The Kissing Problem

- Given a unit center sphere, the maximum number of unit spheres, in n dimensions, can touch or kiss the center sphere?
- General Solutions does not exist.
- Delsarte Method uses linear programming to provide an upper bound on the number of spheres.
- K(8) = 240, K(24) = 196650.
- K(4) = 24: proved using Delsarte Method by Oleg Musin only 3 years ago.
- For other dimensions, lower bounds have been provided.

The Kissing Problem as a Graph Realization

- Can be formulated as a SDP feasibility problem; but SDP solution may not provide proper rank.

$$\begin{aligned} (\mathbf{e}_i - \mathbf{e}_j)^T Y (\mathbf{e}_i - \mathbf{e}_j) &\geq 2, \ \forall i \neq j, \\ \mathbf{e}_i^T Y \mathbf{e}_i &= 2, \ \forall i \end{aligned}$$

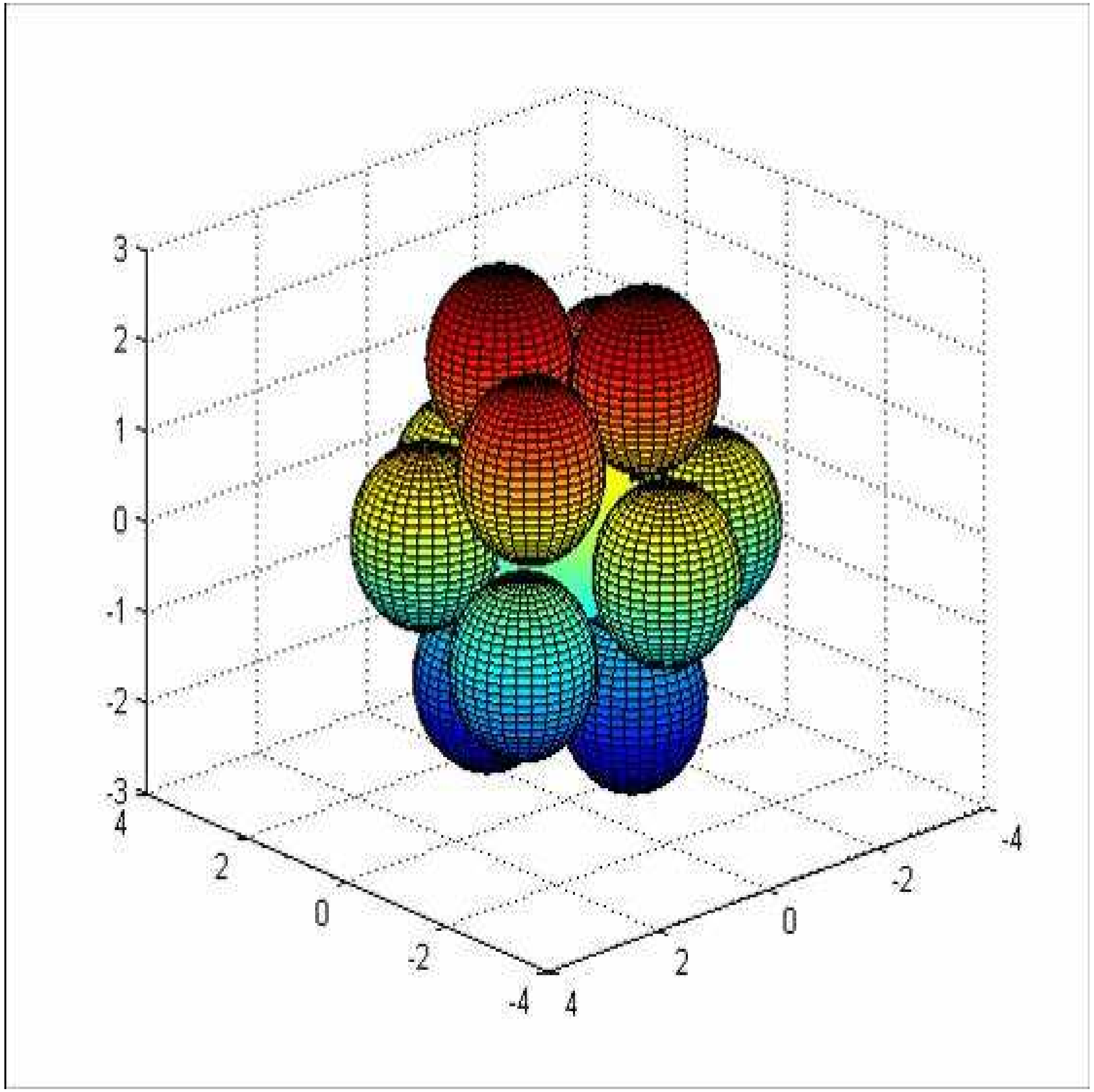
- Construct a nonzero SDP objective:

$$\begin{array}{lrl} \min & C \bullet Y & \\ \text{s.t.} & (\mathbf{e}_i - \mathbf{e}_j)^T Y (\mathbf{e}_i - \mathbf{e}_j) & \geq 2, \ \forall i \neq j, \\ & \mathbf{e}_i^T Y \mathbf{e}_i & = 2, \ \forall i \end{array}$$

Solving the 3-D Kissing Problem

This objective structure can be extended to dimension 3. For 12 spheres, SDP method provides the following realization

Figure 12: 12 Spheres in 3-D



More Applications: Data dimensionality reduction

Given P, a data point set of $\mathbf{p}_1, ..., \mathbf{p}_n \in R^d$, a fundamental question is how to embed P into Q of $\mathbf{q}_1, ..., \mathbf{q}_n \in R^k$, where $k \ll d$, such that $\mathbf{q}_j$s keep all essential information of P, such as the local distances and angles between $\mathbf{p}_j$s.

In other words, find a $d-k$-dimension subspace such that the mapping of $\mathbf{p}_j$s has a minimal "information loss."

More Applications: Molecular confirmation

Anchor-free and 3-D Localization.

More research topics

SDP provides a valubale model for optimization, but more open research topics

- Algorithmic topics: solve SDP faster ...
- How to effectively find a low rank SDP solution ...
- Relaxations on solving other difficult problems: polynomial optimization, ...
- High-order positive semidefinite cones ...