Sensor Localization using Semidefinite Facial Reduction

Nathan Krislock, Henry Wolkowicz

Department of Combinatorics & Optimization
University of Waterloo

Convex Analysis, Optimization and Applications
Les Houches, France
January 8, 2010

Introduction

The Sensor Network Localization (SNL) Problem

Given:

- Positions of some fixed sensors (called anchors)
- Distances between sensors within a fixed radio range

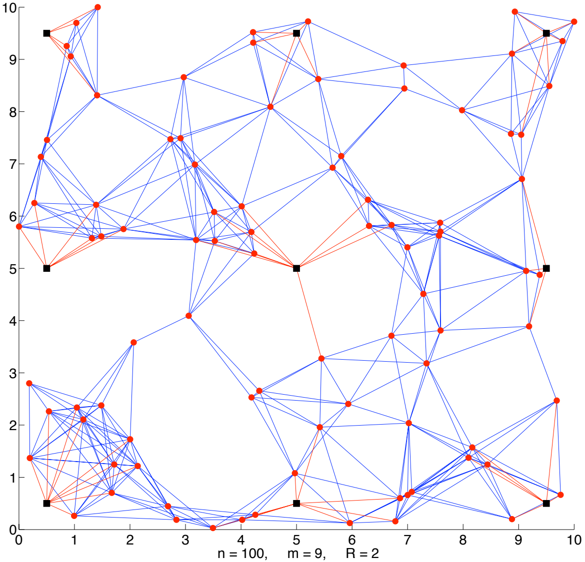
Goal:

- Determine locations of sensors

Motivation

- Many applications use wireless sensor networks:
  - measuring environmental conditions, tracking of goods, random deployment in inaccessible terrains, surveillance, ...
- Problem without anchors is used for molecular conformation

Introduction



Outline

1. Sensor Network Localization (SNL)
   - Introduction
   - Euclidean Distance Matrices and Semidefinite Matrices

Outline

1. Sensor Network Localization (SNL)
   - Introduction
   - Euclidean Distance Matrices and Semidefinite Matrices

2. Facial Reduction
   - Semidefinite Facial Reduction

Outline

1. Sensor Network Localization (SNL)
   - Introduction
   - Euclidean Distance Matrices and Semidefinite Matrices
2. Facial Reduction
   - Semidefinite Facial Reduction
3. Locating Sensors
   - Results
   - Numerical Results

Outline

1. Sensor Network Localization (SNL)
   - Introduction
   - Euclidean Distance Matrices and Semidefinite Matrices
2. Facial Reduction
   - Semidefinite Facial Reduction
3. Locating Sensors
   - Results
   - Numerical Results

Introduction

Notation

- $p_1, \ldots, p_{n-m} \in \mathbb{R}^r$ - unknown points (sensors)
- $a_1, \ldots, a_m \in \mathbb{R}^r$ - known points (anchors)
  - anchors also labeled $p_{n-m+1}, \ldots, p_n$

$$P = \begin{bmatrix} p_1^T \\ \vdots \\ p_n^T \end{bmatrix} = \begin{bmatrix} X \\ A \end{bmatrix} \in \mathbb{R}^{n \times r}$$

- r - embedding dimension (usually 2 or 3)
- $R > 0$ - radio range

Introduction

Graph Realization

- $G = (N, E, w)$ - underlying weighted graph
  - $N = \{1, \ldots, n\}$
  - $w_{ij} = \|p_i - p_j\|$ if $(i, j) \in E$
- SNL problem $\equiv$ find realization of graph in $\mathbb{R}^r$

Euclidean Distance Matrix (EDM) Completion

- $D \in \mathcal{S}^n$ - partial EDM:

$$D_{ij} = \begin{cases} \|p_i - p_j\|^2 & \text{if } (i, j) \in E \\ ? & \text{otherwise} \end{cases}$$

- SNL problem $\equiv$ find EDM completion with embed. dim. $= r$

Outline

1. Sensor Network Localization (SNL)
   - Introduction
   - Euclidean Distance Matrices and Semidefinite Matrices
2. Facial Reduction
   - Semidefinite Facial Reduction
3. Locating Sensors
   - Results
   - Numerical Results

EDMs and Semidefinite Matrices

Linear Transformation $\mathcal{K}$

- Let D be the EDM given by the points $P \in \mathbb{R}^{n \times r}$
- Let $Y := PP^T = (p_i^T p_j)$

EDMs and Semidefinite Matrices

Linear Transformation $\mathcal{K}$

- Let D be the EDM given by the points $P \in \mathbb{R}^{n \times r}$
- Let $Y := PP^T = (p_i^T p_j)$

$$\begin{aligned} D_{ij} &= \|p_i - p_j\|^2 \\ &= p_i^T p_i + p_j^T p_j - 2p_i^T p_j \\ &= Y_{ii} + Y_{jj} - 2Y_{ij} \end{aligned}$$

EDMs and Semidefinite Matrices

Linear Transformation $\mathcal{K}$

- Let D be the EDM given by the points $P \in \mathbb{R}^{n \times r}$
- Let $Y := PP^T = (p_i^T p_j)$

$$\begin{aligned} D_{ij} &= \|p_i - p_j\|^2 \\ &= p_i^T p_i + p_j^T p_j - 2p_i^T p_j \\ &= Y_{ii} + Y_{jj} - 2Y_{ij} \end{aligned}$$

- Thus $D = \mathcal{K}(Y)$, where:

$$\boxed{\mathcal{K}(Y) := \mathrm{diag}(Y)e^T + e\,\mathrm{diag}(Y)^T - 2Y}$$

EDMs and Semidefinite Matrices

Linear Transformation $\mathcal{K}$

- Let D be the EDM given by the points $P \in \mathbb{R}^{n \times r}$
- Let $Y := PP^T = (p_i^T p_j)$

$$\begin{aligned} D_{ij} &= \|p_i - p_j\|^2 \\ &= p_i^T p_i + p_j^T p_j - 2p_i^T p_j \\ &= Y_{ii} + Y_{jj} - 2Y_{ij} \end{aligned}$$

- Thus $D = \mathcal{K}(Y)$, where:

$$\boxed{\mathcal{K}(Y) := \mathrm{diag}(Y)e^T + e\,\mathrm{diag}(Y)^T - 2Y}$$

- $Y = PP^T$ is positive semidefinite, $\mathrm{rank}(Y) = r$
- $\mathcal{K}$ maps the semidefinite cone, $\mathcal{S}_+^n$, onto the EDM cone, $\mathcal{E}^n$

EDMs and Semidefinite Matrices

Vector Formulation

Find $p_1, \ldots, p_n \in \mathbb{R}^r$ such that $\sum_{i=1}^{n} p_i = 0$, $\|p_i - p_j\|^2 = D_{ij}$, $\forall (i,j) \in E$

EDMs and Semidefinite Matrices

Vector Formulation

Find $p_1, \ldots, p_n \in \mathbb{R}^r$ such that $\sum_{i=1}^{n} p_i = 0$, $\|p_i - p_j\|^2 = D_{ij}$, $\forall (i,j) \in E$

Matrix Formulation using $\mathcal{K}$

Find $P \in \mathbb{R}^{n \times r}$ such that $P^T e = 0$, $W \circ \mathcal{K}(Y) = D$, $Y = PP^T$

EDMs and Semidefinite Matrices

Vector Formulation

Find $p_1, \dots, p_n \in \mathbb{R}^r$ such that $\sum_{i=1}^{n} p_i = 0$, $\|p_i - p_j\|^2 = D_{ij}$, $\forall (i,j) \in E$

Matrix Formulation using $\mathcal{K}$

Find $P \in \mathbb{R}^{n \times r}$ such that $P^T e = 0$, $W \circ \mathcal{K}(Y) = D$, $Y = PP^T$

Semidefinite Programming (SDP) Relaxation

Find $Y \in \mathcal{S}^n_+$ s.t. $Ye = 0$, $W \circ \mathcal{K}(Y) = D$

EDMs and Semidefinite Matrices

Vector Formulation

Find $p_1, \ldots, p_n \in \mathbb{R}^r$ such that $\sum_{i=1}^{n} p_i = 0$, $\|p_i - p_j\|^2 = D_{ij}$, $\forall (i,j) \in E$

Matrix Formulation using $\mathcal{K}$

Find $P \in \mathbb{R}^{n \times r}$ such that $P^T e = 0$, $W \circ \mathcal{K}(Y) = D$, $Y = PP^T$

Semidefinite Programming (SDP) Relaxation

Find $Y \in \mathcal{S}^n_+$ s.t. $Ye = 0$, $W \circ \mathcal{K}(Y) = D$

- Vector/Matrix Formulation is non-convex and NP-HARD
- SDP Relaxation is convex, but degenerate (strict feasibility fails)

Outline

1. Sensor Network Localization (SNL)
   - Introduction
   - Euclidean Distance Matrices and Semidefinite Matrices
2. Facial Reduction
   - Semidefinite Facial Reduction
3. Locating Sensors
   - Results
   - Numerical Results

Semidefinite Facial Reduction

An LP Example

$$
\begin{array}{lcccccccccccc}
\text{minimize} & 2x_1 & + & 6x_2 & - & x_3 & - & 2x_4 & + & 7x_5 & & \\
\text{subject to} & x_1 & + & x_2 & + & x_3 & + & x_4 & & & = & 1 \\
 & x_1 & - & x_2 & - & x_3 & & & + & x_5 & = & -1 \\
 & x_1 & , & x_2 & , & x_3 & , & x_4 & , & x_5 & \geq & 0
\end{array}
$$

Semidefinite Facial Reduction

An LP Example

$$
\begin{array}{lccccccccccccc}
\text{minimize} & 2x_1 & + & 6x_2 & - & x_3 & - & 2x_4 & + & 7x_5 & & \\
\text{subject to} & x_1 & + & x_2 & + & x_3 & + & x_4 & & & = & 1 \\
& x_1 & - & x_2 & - & x_3 & & & + & x_5 & = & \text{-1} \\
& x_1 & , & x_2 & , & x_3 & , & x_4 & , & x_5 & \geq & 0
\end{array}
$$

Summing the constraints:

$$2x_1 + x_4 + x_5 = 0 \quad \Rightarrow \quad x_1 = x_4 = x_5 = 0$$

Semidefinite Facial Reduction

An LP Example

$$\begin{array}{lccccccccccc}
\text{minimize} & 2x_1 & + & 6x_2 & - & x_3 & - & 2x_4 & + & 7x_5 & & \\
\text{subject to} & x_1 & + & x_2 & + & x_3 & + & x_4 & & & = & 1 \\
 & x_1 & - & x_2 & - & x_3 & & & + & x_5 & = & -1 \\
 & x_1 & , & x_2 & , & x_3 & , & x_4 & , & x_5 & \geq & 0
\end{array}$$

Summing the constraints:

$$2x_1 + x_4 + x_5 = 0 \quad \Rightarrow \quad x_1 = x_4 = x_5 = 0$$

Reduced LP

$$\begin{array}{lccccc}
\text{minimize} & 6x_2 & - & x_3 & & \\
\text{subject to} & x_2 & + & x_3 & = & 1 \\
 & x_2 & , & x_3 & \geq & 0
\end{array}$$

Semidefinite Facial Reduction

Faces of $\mathbb{R}^n_+ = \{x \in \mathbb{R}^n \ : \ x \geq 0\}$

- A cone $F \subseteq \mathbb{R}^n_+$ is a face of $\mathbb{R}^n_+$ (denoted $F \trianglelefteq \mathbb{R}^n_+$) if

$$x, y \in \mathbb{R}^n_+ \quad \text{and} \quad \frac{1}{2}(x+y) \in F \quad \implies \quad x, y \in F$$

- $F = \left\{x \in \mathbb{R}^5_+ \ : \ x_1 = x_4 = x_5 = 0\right\}$ is a face of $\mathbb{R}^5_+$

Semidefinite Facial Reduction

Faces of $\mathbb{R}^n_+ = \{x \in \mathbb{R}^n \ : \ x \geq 0\}$

- A cone $F \subseteq \mathbb{R}^n_+$ is a face of $\mathbb{R}^n_+$ (denoted $F \trianglelefteq \mathbb{R}^n_+$) if

$$x, y \in \mathbb{R}^n_+ \quad \text{and} \quad \frac{1}{2}(x+y) \in F \quad \implies \quad x, y \in F$$

- $F = \left\{x \in \mathbb{R}^5_+ \ : \ x_1 = x_4 = x_5 = 0\right\}$ is a face of $\mathbb{R}^5_+$

Faces of $\mathcal{S}^n_+ = \{X \in \mathcal{S}^n \ : \ X \succeq 0\}$

- A cone $F \subseteq \mathcal{S}^n_+$ is a face of $\mathcal{S}^n_+$ (denoted $F \trianglelefteq \mathcal{S}^n_+$) if

$$X, Y \in \mathcal{S}^n_+ \quad \text{and} \quad \frac{1}{2}(X+Y) \in F \quad \implies \quad X, Y \in F$$

- If $S \subseteq \mathcal{S}^n_+$, then $\mathrm{face}(S)$ is the smallest face of $\mathcal{S}^n_+$ containing S

Semidefinite Facial Reduction

Representing Faces of $\mathcal{S}^n_+$

If $F \trianglelefteq \mathcal{S}^n_+$ and $X \in \operatorname{relint}(F)$ with $\operatorname{rank}(X) = t$, then

$$F = U\mathcal{S}^t_+ U^T$$

where $X = U\Lambda U^T$ is the compact eigenvalue decomp. with $U \in \mathbb{R}^{n \times t}$

Semidefinite Facial Reduction

Representing Faces of $\mathcal{S}^n_+$

If $F \trianglelefteq \mathcal{S}^n_+$ and $X \in \mathrm{relint}(F)$ with $\mathrm{rank}(X) = t$, then

$$F = U\mathcal{S}^t_+U^T$$

where $X = U\Lambda U^T$ is the compact eigenvalue decomp. with $U \in \mathbb{R}^{n\times t}$

Semidefinite Facial Reduction

- If $\mathrm{face}\left(\left\{X \in \mathcal{S}^n_+ \ : \ \langle A_i, X\rangle = b_i, \ \forall i\right\}\right) = U\mathcal{S}^t_+U^T$, then

$$\begin{array}{ll} \text{minimize} & \langle C, X\rangle \\ \text{subject to} & \langle A_i, X\rangle = b_i, \ \forall i \quad \Longrightarrow \\ & X \in \mathcal{S}^n_+ \end{array}$$

Semidefinite Facial Reduction

Representing Faces of $\mathcal{S}_+^n$

If $F \trianglelefteq \mathcal{S}_+^n$ and $X \in \mathrm{relint}(F)$ with $\mathrm{rank}(X) = t$, then

$$F = U\mathcal{S}_+^t U^T$$

where $X = U\Lambda U^T$ is the compact eigenvalue decomp. with $U \in \mathbb{R}^{n\times t}$

Semidefinite Facial Reduction

- If $\mathrm{face}\left(\left\{X \in \mathcal{S}_+^n \ : \ \langle A_i, X\rangle = b_i, \ \forall i\right\}\right) = U\mathcal{S}_+^t U^T$, then

$$\begin{array}{ll} \text{minimize} & \langle C, X\rangle \\ \text{subject to} & \langle A_i, X\rangle = b_i, \ \forall i \\ & X \in \mathcal{S}_+^n \end{array} \quad \Longrightarrow \quad \begin{array}{ll} \text{minimize} & \langle C, UZU^T\rangle \\ \text{subject to} & \langle A_i, UZU^T\rangle = b_i, \ \forall i \\ & Z \in \mathcal{S}_+^t \end{array}$$

Semidefinite Facial Reduction

Representing Faces of $\mathcal{S}^n_+$

If $F \trianglelefteq \mathcal{S}^n_+$ and $X \in \operatorname{relint}(F)$ with $\operatorname{rank}(X) = t$, then

$$F = U\mathcal{S}^t_+ U^T$$

where $X = U \Lambda U^T$ is the compact eigenvalue decomp. with $U \in \mathbb{R}^{n \times t}$

Semidefinite Facial Reduction

- If $\operatorname{face}\left(\left\{X \in \mathcal{S}^n_+ \; : \; \langle A_i, X\rangle = b_i, \; \forall i\right\}\right) = U\mathcal{S}^t_+ U^T$, then

$$\begin{array}{ll} \text{minimize} & \langle C, X\rangle \\ \text{subject to} & \langle A_i, X\rangle = b_i, \; \forall i \\ & X \in \mathcal{S}^n_+ \end{array} \implies \begin{array}{ll} \text{minimize} & \langle U^T C U, Z\rangle \\ \text{subject to} & \langle U^T A_i U, Z\rangle = b_i, \; \forall i \\ & Z \in \mathcal{S}^t_+ \end{array}$$

Outline

Results

Single Clique Facial Reduction Theorem: (K & Wolkowicz, 2009)

Let:

- D be a partial EDM such that

$$D = \left[\begin{array}{c|c} \bar{D} & \cdot \\ \hline \cdot & \cdot \end{array}\right], \qquad \text{for some } \bar{D} \in \mathcal{E}^k \text{ with embed. dim. } t \leq r$$

- $F := \left\{ Y \in \mathcal{S}^n_+ \cap \mathcal{S}_C : \mathcal{K}(Y[1:k]) = \bar{D} \right\}$ (contains SDP feas. set)

Results

Single Clique Facial Reduction Theorem: (K & Wolkowicz, 2009)

Let:

- D be a partial EDM such that

$$D = \left[\begin{array}{c|c} \bar{D} & \cdot \\ \hline \cdot & \cdot \end{array}\right], \qquad \text{for some } \bar{D} \in \mathcal{E}^k \text{ with embed. dim. } t \leq r$$

- $F := \left\{ Y \in \mathcal{S}^n_+ \cap \mathcal{S}_C : \mathcal{K}(Y[1:k]) = \bar{D} \right\}$ (contains SDP feas. set)

Then:

$$\boxed{\operatorname{face}(F) = \left(U \mathcal{S}_+^{n-k+t+1} U^T \right) \cap \mathcal{S}_C}$$

where $U := \left[\begin{array}{c|c} \bar{U} & 0 \\ \hline 0 & I_{n-k} \end{array}\right]$, $\bar{U} \in \mathbb{R}^{k \times (t+1)}$ eigenvectors of $B := \mathcal{K}^\dagger(\bar{D})$

Results

Algorithm

- For each $i \in N$, find a clique C_i containing node i

Results

Algorithm

- For each $i \in N$, find a clique C_i containing node i
- Let C_{n+1} be the clique of anchors

Results

Algorithm

- For each $i \in N$, find a clique C_i containing node i
- Let C_{n+1} be the clique of anchors
- Let $F_i = \left(U_i \mathcal{S}_+^{n-|C_i|+r+1} U_i^T \right) \cap \mathcal{S}_C$ be the corresponding faces

Results

Algorithm

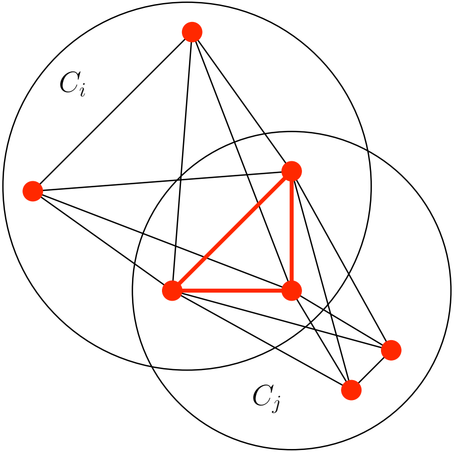
- For each $i \in N$, find a clique C_i containing node i
- Let C_{n+1} be the clique of anchors
- Let $F_i = \left(U_i \mathcal{S}_+^{n-|C_i|+r+1} U_i^T \right) \cap \mathcal{S}_C$ be the corresponding faces
- Compute $F = \bigcap_{i=1}^{n+1} F_i$

Results

Algorithm

- For each $i \in N$, find a clique C_i containing node i
- Let C_{n+1} be the clique of anchors
- Let $F_i = \left(U_i \mathcal{S}_+^{n-|C_i|+r+1} U_i^T\right) \cap \mathcal{S}_C$ be the corresponding faces
- Compute $F = \bigcap_{i=1}^{n+1} F_i$
- Then face $\left\{ Y \in \mathcal{S}_+^n \cap \mathcal{S}_C \; : \; W \circ \mathcal{K}(Y) = D \right\} \subseteq F$

Results



Results

Two Clique Facial Reduction Theorem: (K & Wolkowicz, 2009)

Let $D \in \mathcal{E}^n$ with embed. dim. r. Let $C_1, C_2 \subseteq 1:n$.
For $i = 1, 2$ let:

- $t_i :=$ embed. dim. of $D[C_i]$
- $F_i := \left\{ Y \in \mathcal{S}^n_+ \cap \mathcal{S}_C : \mathcal{K}(Y[C_i]) = D[C_i] \right\}$ (contains SDP feas. set)
- $\mathrm{face}(F_i) =: \left(U_i \mathcal{S}_+^{n-|C_i|+t_i+1} U_i^T \right) \cap \mathcal{S}_C$

Two Clique Facial Reduction Theorem: (K & Wolkowicz, 2009)

Let $D \in \mathcal{E}^n$ with embed. dim. r. Let $C_1, C_2 \subseteq 1:n$.
For $i = 1, 2$ let:

- $t_i :=$ embed. dim. of $D[C_i]$
- $F_i := \left\{ Y \in \mathcal{S}^n_+ \cap \mathcal{S}_C : \mathcal{K}(Y[C_i]) = D[C_i] \right\}$ (contains SDP feas. set)
- $\mathrm{face}(F_i) =: \left(U_i \mathcal{S}_+^{n-|C_i|+t_i+1} U_i^T \right) \cap \mathcal{S}_C$

Then:

$$\boxed{\mathrm{face}(F_1 \cap F_2) = \left(U \mathcal{S}_+^{n-|C_1 \cup C_2|+t+1} U^T \right) \cap \mathcal{S}_C}$$

where $U \in \mathbb{R}^{n \times (t+1)}$ full column rank s.t. $\mathrm{col}(U) = \mathrm{col}(U_1) \cap \mathrm{col}(U_2)$

Results

Subspace Intersection for Two Intersecting Cliques

Suppose:

$$U_1 = \begin{bmatrix} U_1' & 0 \\ U_1'' & 0 \\ 0 & I \end{bmatrix} \quad \text{and} \quad U_2 = \begin{bmatrix} I & 0 \\ 0 & U_2'' \\ 0 & U_2' \end{bmatrix}$$

Then:

$$U := \begin{bmatrix} U_1' \\ U_1'' \\ U_2'(U_2'')^\dagger U_1'' \end{bmatrix} \quad \text{or} \quad U := \begin{bmatrix} U_1'(U_1'')^\dagger U_2'' \\ U_2'' \\ U_2' \end{bmatrix}$$

Satisfies:

$$\operatorname{col}(U) = \operatorname{col}(U_1) \cap \operatorname{col}(U_2)$$

Computing Sensor Positions

Let:

- D be a partial EDM with embed. dim. r
- $F := \left\{ Y \in \mathcal{S}_+^n \cap \mathcal{S}_C : W \circ \mathcal{K}(Y) = D \right\}$ and let $Y \in F$
- $\mathrm{face}(F) =: \left(U \mathcal{S}_+^{r+1} U^T \right) \cap \mathcal{S}_C = (UV) \mathcal{S}_+^r (UV)^T$

Computing Sensor Positions

Let:

- D be a partial EDM with embed. dim. r
- $F := \{Y \in \mathcal{S}^n_+ \cap \mathcal{S}_C : W \circ \mathcal{K}(Y) = D\}$ and let $Y \in F$
- $\operatorname{face}(F) =: \left(U\mathcal{S}^{r+1}_+U^T\right) \cap \mathcal{S}_C = (UV)\mathcal{S}^r_+(UV)^T$

If $D_p[\beta]$ is complete with embed. dim. r then:

- $\mathcal{K}(Y[\beta]) = D_p[\beta]$

Results

Computing Sensor Positions

Let:

- D be a partial EDM with embed. dim. r
- $F := \left\{ Y \in \mathcal{S}^n_+ \cap \mathcal{S}_C : W \circ \mathcal{K}(Y) = D \right\}$ and let $Y \in F$
- $\mathrm{face}(F) =: \left(U \mathcal{S}^{r+1}_+ U^T \right) \cap \mathcal{S}_C = (UV)\mathcal{S}^r_+ (UV)^T$

If $D_p[\beta]$ is complete with embed. dim. r then:

- $\mathcal{K}(Y[\beta]) = D_p[\beta]$
- $Y = (UV)Z(UV)^T$, for some $Z \in \mathcal{S}^r_+$

Computing Sensor Positions

Let:

- D be a partial EDM with embed. dim. r
- $F := \left\{ Y \in \mathcal{S}^n_+ \cap \mathcal{S}_C : W \circ \mathcal{K}(Y) = D \right\}$ and let $Y \in F$
- $\mathrm{face}(F) =: \left(U \mathcal{S}^{r+1}_+ U^T \right) \cap \mathcal{S}_C = (UV)\mathcal{S}^r_+(UV)^T$

If $D_p[\beta]$ is complete with embed. dim. r then:

- $\mathcal{K}(Y[\beta]) = D_p[\beta]$
- $Y = (UV)Z(UV)^T$, for some $Z \in \mathcal{S}^r_+$
- $(JU[\beta,:]V)Z(JU[\beta,:]V)^T = \mathcal{K}^\dagger(D_p[\beta])$ has a unique solution Z

Results

Computing Sensor Positions

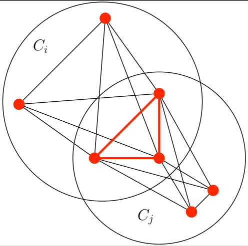
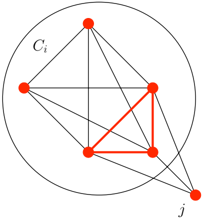
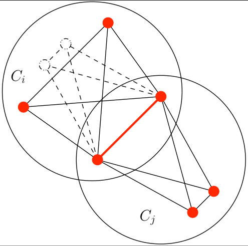
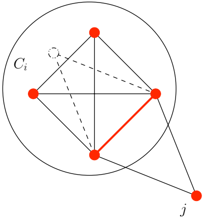
Let:

- D be a partial EDM with embed. dim. r
- $F := \left\{ Y \in \mathcal{S}_+^n \cap \mathcal{S}_C : W \circ \mathcal{K}(Y) = D \right\}$ and let $Y \in F$
- $\mathrm{face}(F) =: \left(U \mathcal{S}_+^{r+1} U^T \right) \cap \mathcal{S}_C = (UV) \mathcal{S}_+^r (UV)^T$

If $D_p[\beta]$ is complete with embed. dim. r then:

- $\mathcal{K}(Y[\beta]) = D_p[\beta]$
- $Y = (UV)Z(UV)^T$, for some $Z \in \mathcal{S}_+^r$
- $(JU[\beta,:]V)Z(JU[\beta,:]V)^T = \mathcal{K}^\dagger(D_p[\beta])$ has a unique solution Z
- $D = W \circ \mathcal{K}\left(PP^T\right)$ where $\boxed{P := UVZ^{\frac{1}{2}} \in \mathbb{R}^{n \times r}}$

Results

	Clique Union	Node Absorption
Rigid		
Non-rigid		

Outline

Numerical Results

- Random *noiseless* problems
- Dimension $r = 2$
- Square region: $[0, 1] \times [0, 1]$
- $m = 4$ anchors
- Using only Rigid Clique Union and Rigid Node Absorption
- Error measure: Root Mean Square Deviation

$$\text{RMSD} = \left(\frac{1}{n} \sum_{i=1}^{n} \| p_i - p_i^{\text{true}} \|^2 \right)^{1/2}$$

- Used MATLAB on a 2.16 GHz Intel Core 2 Duo with 2 GB of RAM

Numerical Results

of Sensors Located

n # sensors \ R	0.07	0.06	0.05	0.04
2,000	2000	2000	2000	1366
6,000	6000	6000	6000	6000
10,000	10000	10000	10000	10000

CPU Seconds

n # sensors \ R	0.07	0.06	0.05	0.04
2,000	1	1	1	2
6,000	3	3	3	3
10,000	6	6	6	6

RMSD (over located sensors)

n # sensors \ R	0.07	0.06	0.05	0.04
2,000	$3e{-}16$	$3e{-}15$	$7e{-}16$	$7e{-}16$
6,000	$4e{-}16$	$4e{-}16$	$4e{-}16$	$6e{-}16$
10,000	$4e{-}16$	$4e{-}16$	$4e{-}16$	$5e{-}16$

Numerical Results

Huge-Scale Problems

# sensors	# anchors	radio range	RMSD	Time	Average Degree
20,000	4	.025	$7e{-16}$	15s	38.5
40,000	4	.02	$1e{-15}$	46s	49.4
60,000	4	.015	$6e{-16}$	1m 54s	41.8
100,000	4	.01	$2e{-15}$	5m 42s	31.1

Size of largest SDP solved

- $n = 100{,}000, \quad m = 1{,}655{,}000$

Summary

- SDP relaxation of SNL is highly degenerate: The feasible set of this SDP is restricted to a low dimensional face of the SDP cone, causing the Slater constraint qualification (strict feasibility) to fail

Summary

- SDP relaxation of SNL is highly degenerate: The feasible set of this SDP is restricted to a low dimensional face of the SDP cone, causing the Slater constraint qualification (strict feasibility) to fail
- We take advantage of this degeneracy by finding explicit representations of the faces of the SDP cone corresponding to unions of intersecting cliques

Summary

- SDP relaxation of SNL is highly degenerate: The feasible set of this SDP is restricted to a low dimensional face of the SDP cone, causing the Slater constraint qualification (strict feasibility) to fail
- We take advantage of this degeneracy by finding explicit representations of the faces of the SDP cone corresponding to unions of intersecting cliques
- Without using an SDP-solver (eg. SeDuMi, SDPA, SDPT3), we quickly compute the exact solution to the large SDP relaxations