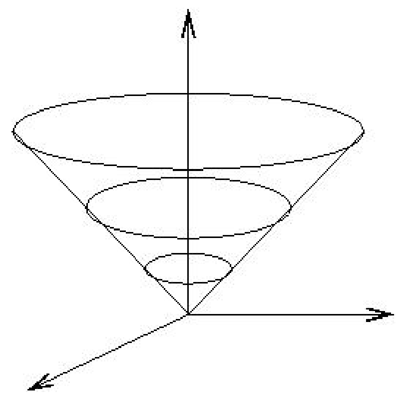
SOCP Relaxation of Sensor Network Localization

Paul Tseng
Mathematics, University of Washington
Seattle

University of Vienna/Wien
June 19, 2006

Talk Outline

- Problem description
- SDP and SOCP relaxations
- Properties of SDP and SOCP relaxations
- Performance of SOCP relaxation and efficient solution methods
- Conclusions & Future Directions



Sensor Network Localization

Basic Problem:

- n pts in $\Re^d$ ($d = 1, 2, 3$).
- Know last $n - m$ pts ('anchors') $x_{m+1}, ..., x_n$ and Eucl. dist. estimate for pairs of 'neighboring' pts

$$d_{ij} \geq 0 \quad \forall (i, j) \in \mathcal{A}$$

with $\mathcal{A} \subseteq \{(i, j) : 1 \leq i < j \leq n\}$.

- Estimate first m pts ('sensors').

History? Graph realization, position estimation in wireless sensor network, determining protein structures, ...

Optimization Problem Formulation

$$v_{\mathrm{opt}} := \min_{x_1,\ldots,x_m} \sum_{(i,j)\in\mathcal{A}} \left| \|x_i - x_j\|^2 - d_{ij}^2 \right|$$

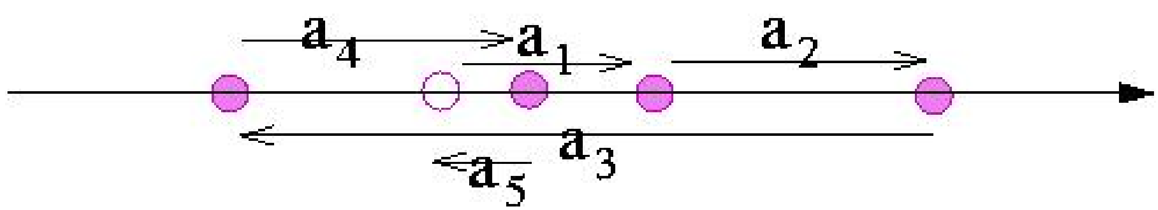
- Objective function is nonconvex.
- Problem is NP-hard (reduction from PARTITION).
- Use a convex (SDP, SOCP) relaxation.

NP-hardness

PARTITION: Given positive integers $a_1, a_2, ..., a_n$, $\exists$ a partition I_1, I_2 of $\{1, ..., n\}$ with $\sum_{i \in I_1} a_i = \sum_{i \in I_2} a_i$? (NP-complete)

Reduction to our problem ($d = 1$) (Saxe '79):

Let $m = n - 1$, $x_n = 0$, $d_{n1} = a_1$, $d_{12} = a_2$, ..., $d_{n-1,n} = a_n$



PARTITION 'yes' $\iff$ $v_{\text{opt}} = 0$

[This extends to $d \geq 2$]

SDP Relaxation

Let $X := [x_1 \ \cdots \ x_m], \quad A := [x_{m+1} \ \cdots \ x_n]$.

Then (Biswas,Ye ’03)

$$\|x_i - x_j\|^2 \quad = \quad \operatorname{tr}\left(b_{ij} b_{ij}^T \begin{bmatrix} X^T X & X^T \\ X & I_d \end{bmatrix} \right)$$

with $b_{ij} := \begin{bmatrix} I_m & 0 \\ 0 & A \end{bmatrix} (e_i - e_j)$.

Fact: $\begin{bmatrix} Y & X^T \\ X & I_d \end{bmatrix} \succeq 0$ has rank $d \iff Y = X^T X$

Thus

$$\begin{aligned} \upsilon_{\text{opt}} = \quad & \min_{X,Y} \quad \sum_{(i,j)\in\mathcal{A}} \left|\operatorname{tr}\left(b_{ij}b_{ij}^T Z\right) - d_{ij}^2\right| \\ & \text{s.t.} \quad Z = \begin{bmatrix} Y & X^T \\ X & I_d \end{bmatrix} \succeq 0, \quad \operatorname{rank} Z = d \end{aligned}$$

Drop low-rank constraint:

$$\boxed{\begin{aligned} \upsilon_{\text{sdp}} := \quad & \min_{X,Y} \quad \sum_{(i,j)\in\mathcal{A}} \left|\operatorname{tr}\left(b_{ij}b_{ij}^T Z\right) - d_{ij}^2\right| \\ & \text{s.t.} \quad Z = \begin{bmatrix} Y & X^T \\ X & I_d \end{bmatrix} \succeq 0 \end{aligned}}$$

- Biswas and Ye gave probabilistic interpretation of SDP soln, and proposed a distributed (domain partitioning) method for solving SDP when $n > 100$.

SOCP Relaxation

Second-order cone program (SOCP) is easier to solve than SDP.

- Q: Is SOCP relaxation a good approximation? Or a mixed SDP-SOCP relaxation?
- Q: How to efficiently solve SOCP relaxation?

SOCP Relaxation

$$
\begin{aligned}
\upsilon_{\text{opt}} = \quad & \min_{x_1,\ldots,x_m,y_{ij}} \quad \sum_{(i,j)\in\mathcal{A}} \left| y_{ij} - d_{ij}^2 \right| \\
& \text{s.t.} \quad y_{ij} = \|x_i - x_j\|^2 \quad \forall (i,j) \in \mathcal{A}
\end{aligned}
$$

Relax "$=$" to "$\geq$" constraint:

$$
\begin{aligned}
\upsilon_{\text{socp}} = \quad & \min_{x_1,\ldots,x_m,y_{ij}} \quad \sum_{(i,j)\in\mathcal{A}} \left| y_{ij} - d_{ij}^2 \right| \\
& \text{s.t.} \qquad y_{ij} \geq \|x_i - x_j\|^2 \quad \forall (i,j) \in \mathcal{A}
\end{aligned}
$$

$$
y \geq \|x\|^2 \quad \Longleftrightarrow \quad y + 1 \geq \|(y-1, 2x)\|
$$

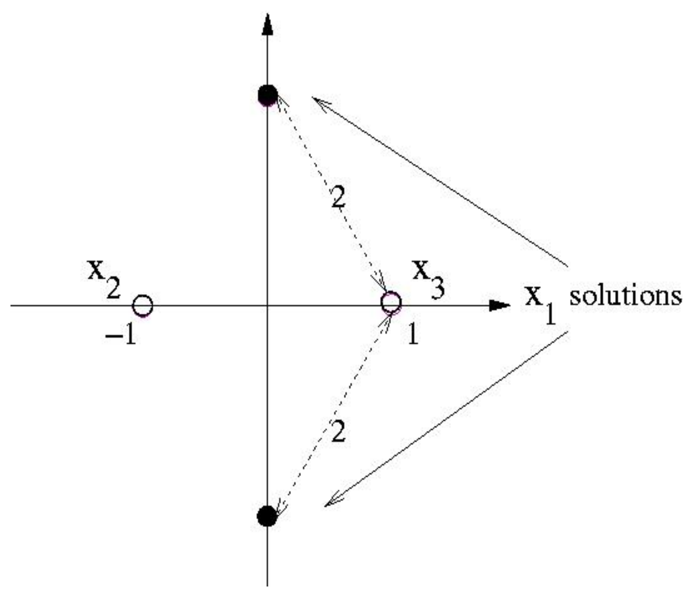
(also Doherty,Pister,El Ghaoui '03)

Properties of SDP, SOCP Relaxations

$d = 2$, $n = 3$, $m = 1$, $d_{12} = d_{13} = 2$

Problem:

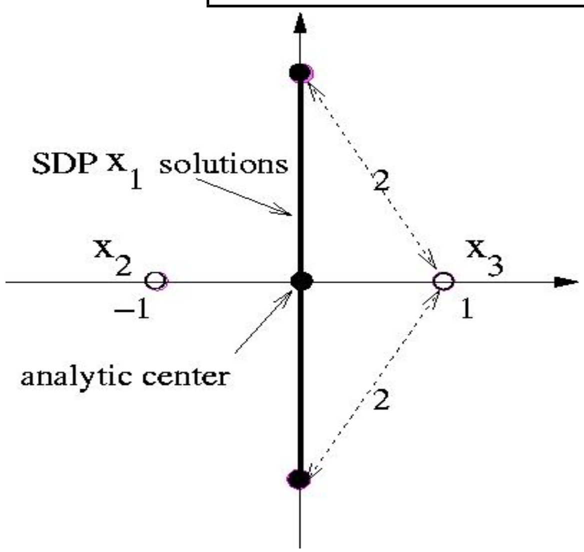
$$0 = \min_{x_1=(\alpha,\beta)\in\Re^2} \quad |(1-\alpha)^2 + \beta^2 - 4| + |(-1-\alpha)^2 + \beta^2 - 4|$$



SDP Relaxation:

$$0 = \min_{\substack{x_1=(\alpha,\beta)\in\Re^2 \\ y\in\Re}} |y - 2\alpha - 3| + |y + 2\alpha - 3|$$

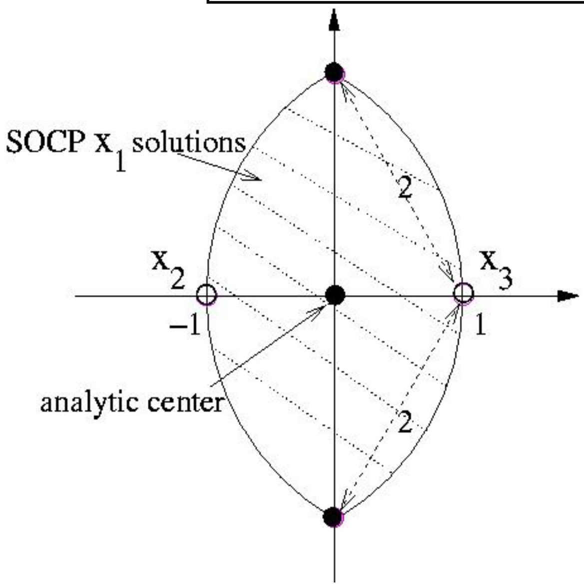
$$\text{s.t.} \quad \begin{bmatrix} y & \alpha & \beta \\ \alpha & 1 & 0 \\ \beta & 0 & 1 \end{bmatrix} \succeq 0$$



If solve SDP by IP method, then likely get analy. center.

SOCP Relaxation:

$$
\begin{aligned}
0 = \quad & \min_{\substack{x_1=(\alpha,\beta)\in\Re^2 \\ y,z\in\Re}} |y-4|+|z-4| \\
\text{s.t.} \quad & y \geq (1-\alpha)^2+\beta^2 \\
& z \geq (-1-\alpha)^2+\beta^2
\end{aligned}
$$



If solve SOCP by IP method, then likely get analy. center.

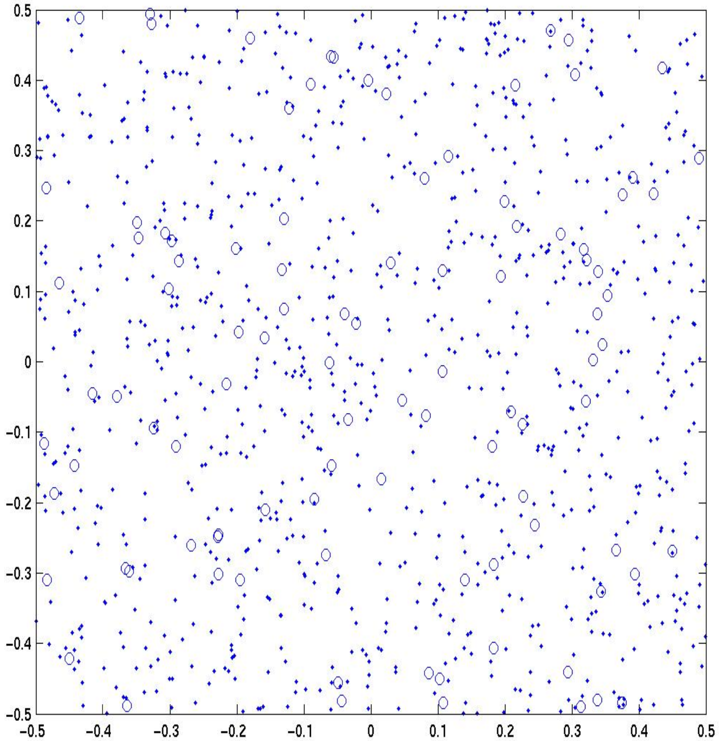
Properties of SDP & SOCP Relaxations

Fact 1: $v_{\rm socp} \le v_{\rm sdp}$. If $v_{\rm socp} = v_{\rm sdp}$, then $\{$SOCP $(x_1, ..., x_m)$ solns$\} \supseteq \{$SDP $(x_1, ..., x_m)$ solns$\}$.

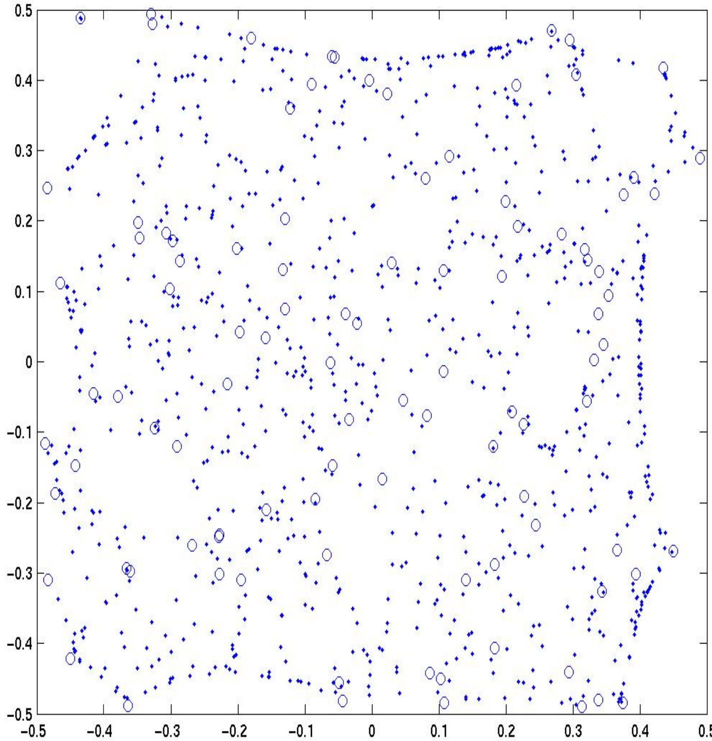
Fact 2: If $(x_1, ..., x_m, y_{ij})_{(i,j)\in\mathcal{A}}$ is the analytic center soln of SOCP, then

$$x_i \in \text{conv}\,\{x_j\}_{j\in\mathcal{N}(i)} \quad \forall i \le m$$

with $\mathcal{N}(i) := \{j \ : \ (i,j) \in \mathcal{A}\}$.



Opt soln ($m = 900$, $n = 1000$, nhbrs if dist$< .06$)



SOCP soln found by IP method (SeDuMi)

Fact 3: If $X = [x_1 \ \cdots \ x_m], Y$ is a relative-interior SDP soln (e.g., analytic center), then for each i,

$$\|x_i\|^2 = Y_{ii} \quad \Longrightarrow \quad x_i \text{ appears in every SDP soln.}$$

Fact 4: If $(x_1, ..., x_m, y_{ij})_{(i,j)\in\mathcal{A}}$ is a relative-interior SOCP soln (e.g., analytic center), then for each i,

$$\|x_i - x_j\|^2 = y_{ij} \quad \text{for some } j \in \mathcal{N}(i) \quad \Longleftrightarrow \quad x_i \text{ appears in every SOCP soln.}$$

Error Bounds

What if distances have errors?

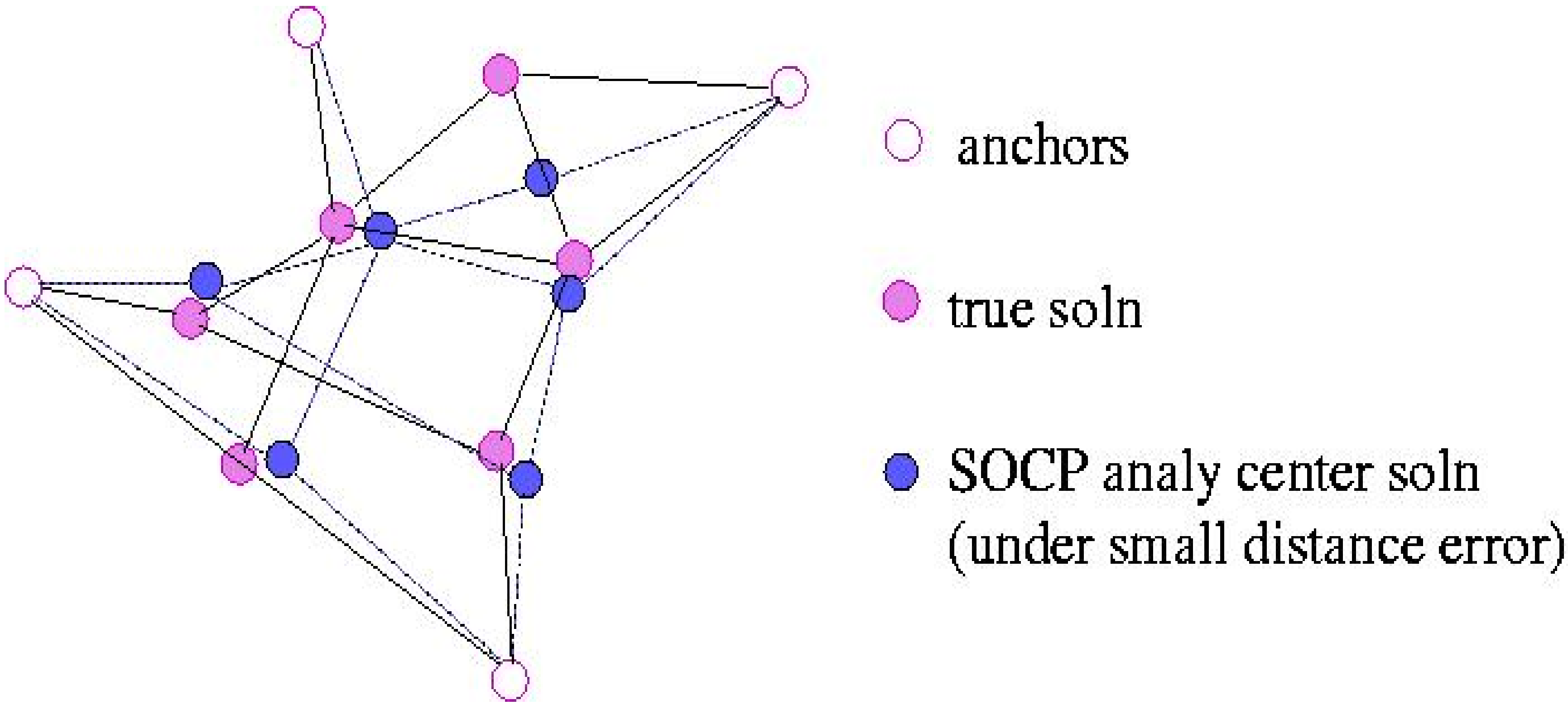
$$d_{ij}^2 = \bar{d}_{ij}^2 + \delta_{ij},$$

where $\delta_{ij} \in \Re$ and $\bar{d}_{ij} := \|x_i^{\text{true}} - x_j^{\text{true}}\|$ ($x_i^{\text{true}} = x_i \ \forall\, i > m$).

Fact 5: If $(x_1, ..., x_m, y_{ij})_{(i,j)\in\mathcal{A}}$ is a relative-interior SOCP soln corresp. $(d_{ij})_{(i,j)\in\mathcal{A}}$ and $\sum_{(i,j)\in\mathcal{A}} |\delta_{ij}| \le \delta$, then for each i,

$$\|x_i - x_j\|^2 = y_{ij} \quad \text{for some } j \in \mathcal{N}(i) \quad \Longrightarrow \quad \|x_i - x_i^{\text{true}}\| = O\Big(\sqrt{\sum_{(i,j)\in\mathcal{A}} |\delta_{ij}|}\Big).$$

Fact 6: As $\sum_{(i,j)\in\mathcal{A}} |\delta_{ij}| \to 0$, (analytic center SOCP soln corresp. $(d_{ij})_{(i,j)\in\mathcal{A}}$) $\to$ (analytic center SOCP soln corresp. $(\bar{d}_{ij})_{(i,j)\in\mathcal{A}}$).



Error bounds for the analytic center SOCP soln when distances have small errors.

Solving SOCP Relaxation I: IP Method

$$\begin{array}{lll} \min\limits_{x_1,\ldots,x_m,y_{ij}} & \sum\limits_{(i,j)\in\mathcal{A}} \left|y_{ij} - d_{ij}^2\right| & \\ \text{s.t.} & y_{ij} \geq \|x_i - x_j\|^2 & \forall (i,j)\in\mathcal{A} \end{array}$$

Put into conic form:

$$\begin{array}{llll} \min & \sum\limits_{(i,j)\in\mathcal{A}} u_{ij} + v_{ij} & & \\ \text{s.t.} & x_i - x_j - w_{ij} & = 0 & \forall (i,j)\in\mathcal{A} \\ & y_{ij} - u_{ij} + v_{ij} & = d_{ij}^2 & \forall (i,j)\in\mathcal{A} \\ & \alpha_{ij} & = \frac{1}{2} & \forall (i,j)\in\mathcal{A} \\ & u_{ij} \geq 0,\ v_{ij} \geq 0, & (\alpha_{ij}, y_{ij}, w_{ij}) \in \mathrm{Rcone}^{d+2} & \forall (i,j)\in\mathcal{A} \end{array}$$

with $\mathrm{Rcone}^{d+2} := \{(\alpha, y, w) \in \Re \times \Re \times \Re^d : \|w\|^2/2 \leq \alpha y\}$.

Solve by an IP method, e.g., SeDuMi 1.05 (Sturm '01).

Solving SOCP Relaxation II: Smoothing + Coordinate Gradient Descent

$$\min_{y \geq z} |y - d^2| = \max\{0, z - d^2\}$$

So SOCP relaxation:

$$\min_{x_1,\ldots,x_m} \sum_{(i,j)\in\mathcal{A}} \max\{0, \|x_i - x_j\|^2 - d_{ij}^2\}$$

This is an unconstrained nonsmooth convex program.

- Smooth approximation:

$$\max\{0, t\} \approx \mu h(t/\mu) \quad (\mu > 0)$$

h smooth convex, $\lim_{t\to-\infty} h(t) = \lim_{t\to\infty} h(t) - t = 0$. We use $h(t) = ((t^2+4)^{1/2} + t)/2$ (CHKS).

SOCP approximation:

$$\min \quad f_\mu(x_1,..,x_m) := \sum_{(i,j)\in\mathcal{A}} \mu h\left(\frac{\|x_i - x_j\|^2 - d_{ij}^2}{\mu}\right)$$

Add a smoothed log-barrier term $-\mu \sum_{(i,j)\in\mathcal{A}} \log\left(\mu h\left(\frac{d_{ij}^2 - \|x_i - x_j\|^2}{\mu}\right)\right)$

Solve the smooth approximation using coordinate gradient descent (SCGD):

- If $\|\nabla_{x_i} f_\mu\| = \Omega(\mu)$, then update x_i by moving it along the Newton direction $-[\nabla^2_{x_i x_i} f_\mu]^{-1}\nabla_{x_i} f_\mu$, with Armijo stepsize rule, and re-iterate.
- Decrease μ when $\|\nabla_{x_i} f_\mu\| = O(\mu)\ \forall i$.

$\mu^{\text{init}} = 1e-5$. $\mu^{\text{end}} = 1e-9$. Decrease μ by a factor of 10.

Code in Fortran. Computation easily distributes.

Simulation Results

- Uniformly generate $x_1^{\text{true}}, ..., x_n^{\text{true}}$ in $[0, 1]^2$, $m = .9n$,
  two pts are nhbrs if dist$<$ $radiorange$.
  Set
  $$d_{ij} = \|x_i^{\text{true}} - x_j^{\text{true}}\| \cdot \max\{0, 1 + \epsilon_{ij} \cdot nf\},$$
  $\epsilon_{ij} \sim N(0, 1)$ (Biswas, Ye '03)

- Solve SOCP using SeDuMi 1.05 or SCGD.

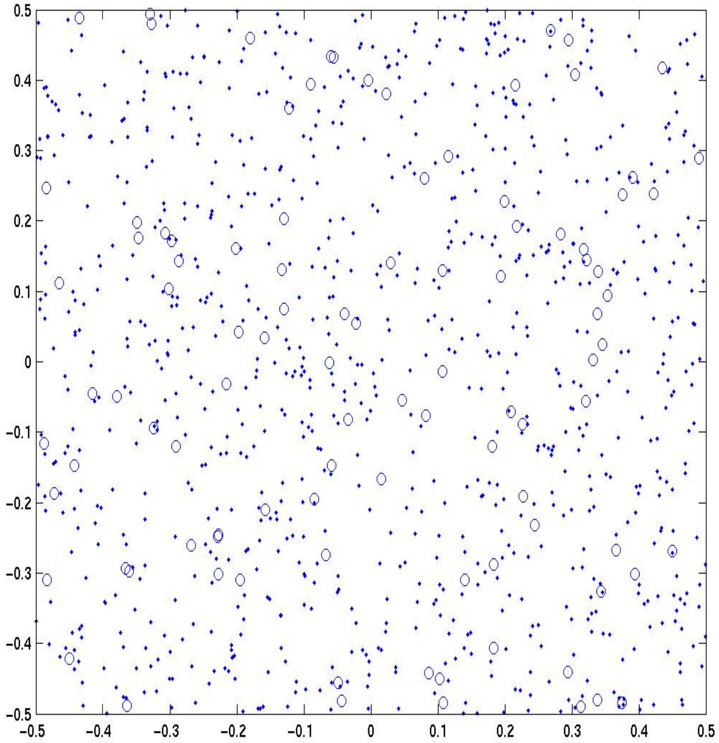
- Sensor i is uniquely positioned if

$$\left| \|x_i - x_j\|^2 - y_{ij} \right| \leq 10^{-7} d_{ij} \quad \text{for some } j \in \mathcal{N}(i).$$

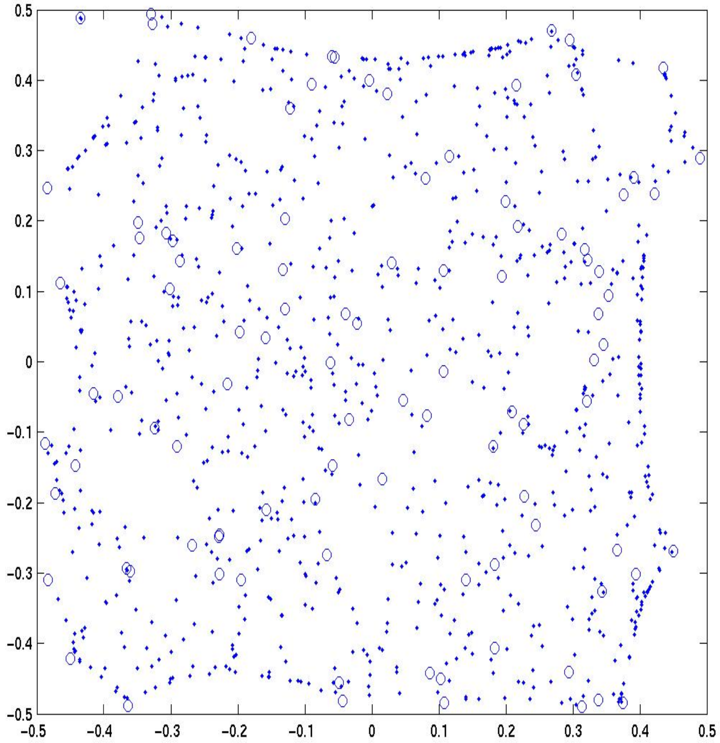
n	m	nf	SeDuMi cpu/$m_{\rm up}$/Err$_{\rm up}$	SCGD cpu/$m_{\rm up}$/Err$_{\rm up}$
1000	900	0	5.5/402/7.2e-4	.4/365/3.7e-5
1000	900	.001	5.4/473/1.8e-3	3.3/451/1.5e-3
1000	900	.01	5.6/554/1.5e-2	2.2/518/1.1e-2
2000	1800	0	209.6/1534/4.3e-4	1.3/1541/3.3e-4
2000	1800	.001	230.1/1464/3.6e-3	6.8/1466/3.6e-3
2000	1800	.01	176.6/1710/5.1e-2	3.7/1710/5.1e-2
4000	3600	0	203.1/2851/4.0e-4	2.5/2864/3.2e-4
4000	3600	.001	205.2/2938/3.2e-3	15.1/2900/3.0e-3
4000	3600	.01	201.3/3073/1.0e-2	23.2/3033/9.1e-3

Table 1: $radiorange = .06(.035)$ for $n = 1000, 2000(4000)$

- cpu (sec) times are on a HP DL360 workstation, running Linux 3.5.
- $m_{\rm up} :=$ number of uniquely positioned sensors.
- Err$_{\rm up} := \max_{i \text{ uniq. pos.}} \|x_i - x_i^{\rm true}\|$.



True positions of sensors (dots) and anchors (circles) ($m = 900$, $n = 1000$)



SOCP soln found by SeDuMi and SCGD

Mixed SDP-SOCP Relaxation

Choose $0 \leq \ell \leq m$. Let $\mathcal{B} := \{(i,j) \in \mathcal{A} \ : \ i \leq \ell, j \leq \ell\}$.

$$\min_{x_1,\ldots,x_m,y_{ij},Y} \quad \sum_{(i,j)\in\mathcal{B}} \left|\mathrm{tr}\left(b_{ij}b_{ij}^T Z\right) - d_{ij}^2\right| + \sum_{(i,j)\in\mathcal{A}\setminus\mathcal{B}} \left|y_{ij} - d_{ij}^2\right|$$

s.t.

$$Z = \begin{bmatrix} Y & X^T \\ X & I_d \end{bmatrix} \succeq 0$$

$$X = [x_1 \ \cdots \ x_\ell]$$

$$y_{ij} \geq \|x_i - x_j\|^2 \quad \forall (i,j) \in \mathcal{A} \setminus \mathcal{B}$$

with $b_{ij} := \begin{bmatrix} I_\ell & 0 & 0 \\ 0 & 0 & A \end{bmatrix} (e_i - e_j)$.

Easier to solve than SDP? As good a relaxation? Properties?

Conclusions & Future Directions

- SOCP relaxation may be a good pre-processor.
- Faster methods for solving SOCP? Exploiting network structures of SOCP? (For $d = 1$, solvable by ϵ-relaxation method (Bertsekas,Polymenakos,T '97))
- Error bound for SDP relaxation?
- Additional (convex) constraints? Other objective functions, e.g., $\sum_{(i,j)\in\mathcal{A}} \left| \|x_i - x_j\| - d_{ij} \right|^2$?
- Replace 2-norm by a p-norm ($1 \leq p \leq \infty$)? p-order cone relaxation?
- Q: For any $x_1, ..., x_n \in \Re^d$, does $\arg\min_x \sum_{i=1}^{n} \|x - x_i\|_p^p \in \operatorname{conv}\{x_1, ..., x_n\}$? ($1 < p < \infty$)
  A: Yes for $d \leq 2$. No for $d \geq 3$.