Semidefinite Programming
Applications and Implementations

Henry Wolkowicz

Dept. of Combinatorics and Optimization
University of Waterloo

CORS/MOPGP
Niagara Falls, June 11, 2012

Outline

Background

Historical Events

- Lyapunov (stability) 1890
- SDP (cone optimization/duality) 1960's
- Engineering applications 60's
- matrix completion problems 80's
- polynomial time algorithms Nesterov-Nemirovski 80's
- combinatorial appl., primal-dual interior-point (p-d i-p) algorithms (explosion of activity) 90's
- sparsity/special-structure/low-rank/large-scale/robust-opt. 00's

What are SDPs ?

Primal and Dual SDPs (look like LPs with matrix variables)

$$\text{(PSDP)} \quad \left\{ \begin{array}{llll} p^* := & \max & \operatorname{trace} CX & (= \langle C, X \rangle) \\ & \text{s.t.} & \mathcal{A}X = b & (b_i = \langle A_i, X \rangle) \\ & & X \succeq 0. & \end{array} \right.$$

$$\text{(DSDP)} \quad \left\{ \begin{array}{llll} d^* := & \min & b^T y & (= \langle b, y \rangle) \\ & \text{s.t.} & \mathcal{A}^* y - Z = C & (\mathcal{A}^* y = \sum_{i=1}^m y_i A_i) \\ & & Z \succeq 0, & \end{array} \right.$$

$\mathcal{S}^n$ space of $n \times n$ real symmetric matrices, $A_i, X, Z, C \in \mathcal{S}^n$
$\succ 0$ ($\succeq 0$) pos. (semi)definiteness; (Loewner partial order)
$\mathcal{A} : \mathcal{S}^n \to \Re^m$ lin. transf.; $\mathcal{A}^*$ adjoint transf. (transpose)

Duality: Primal-Dual Pair PSDP, DSDP

PSDP

$$p^* := \max \{\operatorname{trace} CX : \mathcal{A}X = b, X \succeq 0\}$$

Weak Duality (using hidden constraints)

$$p^* = \max_{X\succeq 0} \min_y \operatorname{trace} CX + [y^T(b - \mathcal{A}X)] \quad \text{(PSDP)}$$

$$\le \min_y \max_{X\succeq 0} y^T b + [\operatorname{trace} \overbrace{\overbrace{(C - \mathcal{A}^* y)}^{Z} X}^{\text{duality gap}}] \quad \text{(best LB)}$$

$$= \min_{C - \mathcal{A}^* y \preceq 0} y^T b =: d^* \quad \text{(DSDP)}$$

$\mathcal{A}^*$ adjoint of $\mathcal{A}$

$$\langle \mathcal{A}(X), y\rangle = y^T \mathcal{A}X = \langle X, \mathcal{A}^* y\rangle = \operatorname{trace} X(\mathcal{A}^* y), \qquad \forall X, \forall y$$

Characterization of (p-d) Optimality

Characterization of Optimality for $Z, X \succeq 0$

$$(*) \left\{ \begin{array}{rcl} \mathcal{A}^* y - Z - C & = & 0 \quad \text{dual feasibility} \\ b - \mathcal{A}(X) & = & 0 \quad \text{primal feasibility} \end{array} \right.$$

$$(**) \left\{ \; ZX = 0 \quad \text{complementary slackness} \right.$$

$X, (y, Z)$ a primal-dual optimal pair; Z (dual) slack variable

Perturbed complementary slackness

For primal-dual interior-point (p-d i-p) methods, replace (**) with
(***) $ZX = \mu I, \quad Z, X \succ 0, \mu > 0$
solve (*) and (***): X_μ, y_μ, Z_μ on Central Path; $\mu \downarrow 0$

Difference with LP

$Z, X \in \mathcal{S}^n$ but ZX is not necessarily symmetric!

Interior-point; path-following $\mu \downarrow 0$

Solve overdetermined linearized system

ZX is not necessarily symmetric; equations (*) and (***) are overdetermined. A possible symmetrization; start with feasible (X, y, Z); set $\mu = \frac{1}{2n}\text{trace}\, ZX$; use linearized system.

1. $\mathcal{A}\Delta X = 0$
2. $\Delta Z = \mathcal{A}^*\Delta y$
3. $Z\Delta X + \Delta Z X = \mu I - ZX$

Eliminate ΔZ and then ΔX; solve for Δy

$\Delta X = \mu Z^{-1} - X - Z^{-1}\mathcal{A}^*(\Delta y)X$ from Items 2, 3
substit. for ΔX in Item 1; then solve for Δy in
$\mathcal{A}(Z^{-1}\mathcal{A}^*(\Delta y)X) = \mu\mathcal{A}(Z^{-1}) - b$

New point

Backsolve to keep $X, Z \succ 0$

- $X \leftarrow X + t\frac{1}{2}(\Delta X + \Delta X^T)$
- $y \leftarrow y + t\Delta y$
- $Z \leftarrow Z + t\Delta Z$

Convergence

Given $\epsilon > 0$; if μ is chosen properly at each iteration, then a full Newton step $t = 1$ is feasible; a duality gap $b^T y - \operatorname{trace} CX < \epsilon$ can be obtained in $O(\sqrt{n}|log \epsilon|)$ iterations.
If the problem is feasible, then there exists an optimum on the boundary.

(unlike LP) Strong Duality Can Fail for SDP

Strong Duality for PSDP

- zero duality gap:

$$p^* = d^*$$

- AND d^* is attained.
- (in the case of attainment for both; X, y, Z feasible:)

$$p^* = d^* \text{ iff } \langle Z, X \rangle = 0 \text{ iff } ZX = 0$$

Regularization using Faces

ref. Borwein-W/80, Ramana/97,Ramana-Tuncel-W/97

Faces of Cones

Face

A convex cone F is a face of K, denoted $F \trianglelefteq K$, if

$$x, y \in K \text{ and } x + y \in F \implies x, y \in F.$$

If $F \trianglelefteq K$ and $F \neq K$, write $F \triangleleft K$.

Conjugate Face

If $F \trianglelefteq K$, the conjugate face (or complementary face) of F is

$$F^c := F^\perp \cap K^* \trianglelefteq K^*,$$

where $K^* = \{\phi : \langle \phi, k \rangle \geq 0, \forall k \in K\}$ (dual/polar cone)
If $x \in \operatorname{relint}(F)$, then $F^c = \{x\}^\perp \cap K^*$.

Faces of SDP Cone

Face $F \unlhd \mathcal{S}^n_+$ Characterized by any $X \in \operatorname{relint} F$

$$X = UDU^T \in \operatorname{relint} F \unlhd \mathcal{S}^n_+, U^T U = I_t, D \in \mathcal{S}^t_{++}$$

iff

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

Conjugate Face of $F \unlhd \mathcal{S}^n_+$

the conjugate face (or complementary face) of F is

$$F^c := F^\perp \cap \mathcal{S}^n_+ = V\mathcal{S}^{n-t}_+ V^T, \quad V^T U = 0, V^T V = I_{n-t}$$

Minimal Face (Minimal Cone)

Feasible set of DSDP

Let $\mathcal{F}_D := \{y : Z = \mathcal{A}^* y - C \succeq 0\}$

Minimal Face

Assume $\mathcal{F}_D$ is nonempty, the minimal face (or minimal cone) of DSDP is

$$f_D := \bigcap \{F \unlhd K : \mathcal{A}^*(\mathcal{F}_D) - C \subset F\}$$

i.e., the minimal face that contains all the feasible slacks.

DSDP for Example from Ramana, 1995

DSDP (Max instead of Min)

$$0 = d^* = \max_y \left\{ y_2 : \begin{pmatrix} y_2 & 0 & 0 \\ 0 & y_1 & y_2 \\ 0 & y_2 & 0 \end{pmatrix} \preceq \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \right\}$$

$$y^* = \begin{pmatrix} y_1^* & 0 \end{pmatrix}^T, \quad y_1^* \leq 0, \quad Z^* = C - \mathcal{A}^* y^* = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -y_1^* & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Constraint Qualification (CQ) Fails

Slater's CQ (strict feasibility) fails for dual

PSDP for Example from Ramana, 1995

Primal Program, PSDP (Min instead of Max)

$$1 = p^* = \min_{X \succeq 0} \quad \{X_{11} : \quad \operatorname{trace} A_1 X = X_{22} = 0, \quad \operatorname{trace} A_2 X = X_{11} + 2X_{23} = 1\}$$

$$X^* = \begin{pmatrix} 1 & 0 & X_{13} \\ 0 & 0 & 0 \\ X_{13} & 0 & X_{33} \end{pmatrix}, \quad X_{33} \geq (X_{13}^2)$$

Slater's CQ for (primal) dual & complementarity fails

<u>duality gap</u> $= p^* - d^* = 1 - 0 = \underline{1 > 0}$,

$$\operatorname{trace} X^* Z^* = \operatorname{trace} \begin{pmatrix} 1 & 0 & X_{13} \\ 0 & 0 & 0 \\ X_{13} & 0 & X_{33} \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & -y_1^* & 0 \\ 0 & 0 & 0 \end{pmatrix} = \underline{1 > 0}$$

Minimal Face for Ramana Example

Feasible Set/Minimal Face

$\mathcal{F}_D = \{y \in \mathbb{R}^2 : y_1 \leq 0,\ y_2 = 0\}$

$$\begin{array}{rcl} f_D & = & \bigcap\{F \trianglelefteq \mathcal{S}^3_+ : C - \mathcal{A}^*(\mathcal{F}_D) \subset F\} \\ & = & \begin{pmatrix} \mathcal{S}^2_+ & 0 \\ 0 & 0 \end{pmatrix} \\ & \lhd & \mathcal{S}^3_+ \end{array}$$

Slater CQ and Minimal Face

If DSDP is feasible, then

$C - \mathcal{A}^*y \not\succ_K 0, \forall y$ (Slater's CQ fails for DSDP) $\iff f_D \lhd K$

Regularization of DSDP

Borwein-W (1981)

If d^* is finite, then DSDP is equivalent to regularized DSDP

$$d^*_{RD} = \max_y \{\langle b, y\rangle \ : \ \mathcal{A}^* y \preceq_{f_D} C\}. \qquad \text{(RD)}$$

Lagrangian Dual DRD Satisfies Strong Duality:

$$d^* = d^*_{RD} = d^*_{DRD} = \min_X \{\langle C, X\rangle \ : \ \mathcal{A}X = b, \ X \succeq_{f_D^*} 0\} \quad \text{(DRD)}$$

and d^*_{DRP} is attained

Implementation Problems with Regularization; but, Many Applications

Difficulties

Borwein and W. also gave an algorithm to compute f_D.
But Difficulties:

1. The algorithm requires the solution of several (homogeneous) cone programs (constraints are: $\mathcal{A}x = 0, \langle c, x \rangle = 0, 0 \neq x \succeq_K 0$)
2. If Slater's CQ fails for PSDP then it also fails for each of these cone programs.

Application to Combinatorial Problems

Slater CQ fails for many applications to combinatorial problems.
But, f_D can be found explicitly.

Further Differences with LP

Strict Complementarity can Fail

$\boxed{Z + X \succ 0}$ Theorem of Goldman and Tucker for LP can fail, though conditions hold generically; ref. Shapiro/99, Pataki-Tuncel/98, Alizadeh-Haeberly-Overton/98)

Polynomial Time Complexity/Algorithms

SDP are convex programs; can be approximately solved in polynomial time by interior point algorithms (ref. Nesterov-Nemirovski/88)

Strong Relaxations of Computationally Hard Problems

Modelling Computationally Hard Problems

- Many computationally hard problems can be modelled as quadratically constrained quadratic programs, (QQP) (rather than LPs).
- QQPs are themselves computationally hard.
- But, Lagrangian relaxation can be solved efficiently using SDP.

Applications

statistics, engineering, matrix completions, approximation theory, nonlinear programming, Euclidean distance matrix completion, (EDM); sensor network localiz. (SNL)
combinatorial optimization: max-cut; graph partitioning; quadratic assignment problem; graph colouring; max-clique.

SDP Software Webpage

Software

List and references available at:
www-user.tu-chemnitz.de/ helmberg/sdp_software.html

- SDPLIB SDPLIB is a collection of semidefinite programming test problems. (in SDPA sparse format)
- CVS, Disciplined Convex Programming
- Solvers: CSDP (exploits BLAS); SeDuMi1.1 (dependable, popular); SDPT3(including quadratic/sensor localization); SDPA (including parallel); GloptiPoly-3 (moments; optimization; and SDP); PENNON (nonlinear SDP); SBmethod(first order method/large scale);

Notes on Software

"The State-of-the-Art in Conic Optimization Software", Hans Mittlemann

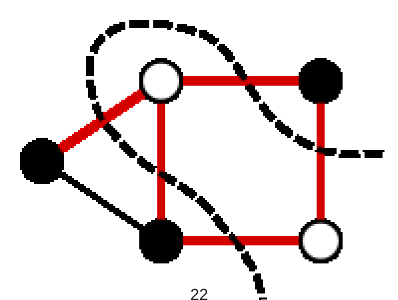
in recent "Handbook on Semidefinite, Conic and Polynomial Optimization" editors Anjos and Lasserre

SDP Relaxation of Max-Cut Problem, (MC)

Max-Cut Problem

undirected, complete, graph $\mathcal{G} = (V, E)$, $|V| = n$, with edge weights w_{ij}; divide nodes into two sets to maximize the sum of weights of cut edges.

A Maximum Cut



Quadratic-Quadratic (QQP) Model for MC

Quadratic Model of MC with Integer Constraints

$$\max \ \frac{1}{2}\sum_{i<j} w_{ij}(1-x_i x_j), \quad x \in \{\pm 1\}^n.$$

Equate $x_i = 1$ with $i \in \mathcal{I}$; and -1 otherwise.

QQP Model of MC

Let L be the Laplacian of $\mathcal{G}$, e.g. if weights are $0,1$

$$L_{ij} = \begin{cases} \deg(v_i) & \text{if } i = j \\ -1 & \text{if } i \neq j \text{ and } v_i \text{ is adjacent to } v_j \\ 0 & \text{otherwise} \end{cases}$$

Let $q(x) := (\frac{1}{4})x^T Lx$; equivalent QP problem

$$(4)\, p^* := \max \left\{ q(x) = x^T Lx : \ x \in \{\pm 1\}^n \right\}$$

SDP Relaxation; use commutativity $\operatorname{trace} AB = \operatorname{trace} BA$

Direct Relaxation

$$(4) \quad p^* := \max \left\{ x^T L x : x \in \{\pm 1\}^n \right\}$$

Replace $x \in \{\pm 1\}^n$ with $x_i^2 = 1$. Note that

$$x^T L x = \operatorname{trace} x^T L x = \operatorname{trace} L x x^T = \operatorname{trace} LX, \text{ with } \overbrace{X = xx^T}^{\text{rank-1}}$$

also: $\quad X \succeq 0, \quad \operatorname{diag}(X) = e, \quad q(x) = \operatorname{trace} LX$

Relax the hard rank-1 condition on X; get SDP relaxation

The SDP Relaxation of MC

$$\boxed{p^* := \max \{ \operatorname{trace} LX : \operatorname{diag}(X) = e, X \succeq 0 \}}$$

Duality for SDP Relaxation of MC

Primal-Dual Programs

$$\text{(PSDP)} \qquad \begin{array}{rll} d^* = p^* := & \max & \operatorname{trace} LX \\ & \text{s.t.} & \operatorname{diag}(X) = e \\ & & X \succeq 0, X \in \mathcal{S}^n, \end{array}$$

diag : vector from diagonal; Diag : diagonal matrix from vector; e vector of ones;

$$\text{(DSDP)} \qquad \begin{array}{rll} p^* = d^* := & \min & e^T y \\ & \text{s.t.} & \operatorname{Diag}(y) - Z = L \\ & & Z \succeq 0, Z \in \mathcal{S}^n, \end{array}$$

Slater Points

$\hat{X} = I \succ 0$; $\qquad \hat{Z} = L - \operatorname{Diag}(\hat{y}) \succ 0$ for $\hat{y} << 0$

Modern Optimality Framework

(Perturbed) Overdetermined Optimality Conditions, $X, Z \succ 0$

$$F_\mu(X,y,Z) = \begin{cases} R_d := \mathrm{Diag}(y) - Z - L & = 0 \quad \text{dual feas.} \\ R_p := \mathrm{diag}(X) - e & = 0 \quad \text{primal feas.} \\ ZX & = 0 \quad \text{compl. slack.} \\ R_c := ZX - \mu I & = 0 \quad \text{pert. compl. slack.} \end{cases}$$

ZX NOT nec. symmetric

Linearization/(Gauss)-Newton Direction

$$F'_\mu(X,y,Z)\begin{pmatrix}\Delta X \\ \Delta y \\ \Delta Z\end{pmatrix} = \begin{bmatrix}\mathrm{Diag}(\Delta y) - \Delta Z \\ \mathrm{diag}(\Delta X) \\ Z\Delta X + \Delta Z X\end{bmatrix} = -F_\mu(X,y,Z)$$

Simple/Efficient Algorithm

Block Eliminations; Block Backsolves

- $\leftarrow$ solve for $\Delta Z = \mathrm{Diag}\,(\Delta y) + R_d$
- substitute $Z\Delta X + (\mathrm{Diag}\,(\Delta y) + R_d)X$
- $\leftarrow$ solve for $\Delta X = Z^{-1}(-\mathrm{Diag}\,(\Delta y)X - R_d X - R_c)$
- substitute and solve for Δy
  $\mathrm{diag}\,\left[Z^{-1}(-\mathrm{Diag}\,(\Delta y)X - R_d X - R_c)\right] = -R_b$
  equivalently $\boxed{\mathrm{diag}\,\left[Z^{-1}\mathrm{Diag}\,(\Delta y)X\right] = (\mu\mathrm{diag}\,(Z^{-1}) - e)}$
  $-\mathrm{diag}\,\left(Z^{-1}R_d X\right) = 0$, since $R_d = 0$ easy to obtain.

Cheat/Symmetrize ΔX in Backsolve; HKM Search Direction

$\leftarrow$ backsolve for $\Delta Z, \Delta X$; $\Delta X \leftarrow \frac{1}{2}(\Delta X + \Delta X^T)$

MATLAB Code

Initialization: $X, Z \succ 0$

```
function [phi, X, y] = psd_ip( L);
% solves: max trace(LX) s.t. X psd, diag(X) = b;  b = ones(n,1)/4
%         min b'y       s.t. Diag(y) - L psd, y unconstrained,
%**input:  L ... symmetric matrix
%**output: phi ... optimal value of primal, phi =trace(LX)
%          X   ... optimal primal matrix
%          y   ... optimal dual vector
% call:   [phi, X, y] = psd_ip( L);
%%%%%%%%%%%%%%%%%%%
         %%Initialization
digits = 6;                     % 6 significant digits of phi
[n, n1] = size( L);             % problem size
b = ones( n,1 ) / 4;            % any b>0 works just as well
X = diag( b);                   % initial primal matrix is pos. def.
y = sum( abs( L))' * 1.1;       % initial y is chosen so that
Z = diag( y) - L;               % initial dual slack Z is pos. def.
phi = b'*y;                     % initial dual costs
psi = L(:)' * X( :);            % and initial primal costs
mu = Z( :)' * X( :)/( 2*n);     % initial complementarity
iter=0;                         % iteration count
```

Find Search Direction/Symmetrize dX

Solve: dy; backsolve: dZ, dX; symmetrize dX

```
disp(['      iter     alphap     alphad      gap      lower      upper']);

while phi-psi > max([1,abs(phi)]) * 10^(-digits)

      iter = iter + 1;           % start a new iteration
      Zi = inv( Z);              % inv(Z) is needed explicitly
      Zi = (Zi + Zi')/2;
      dy =  (Zi.*X) \ (mu * diag(Zi) - b);      % solve for dy
      dX = - Zi * diag( dy) * X + mu * Zi - X;  % back substitute for dX
      dX = ( dX + dX')/2;        % symmetrize
```

Line Search to Stay Interior; and Update

Backtrack to keep $X, Z \succ 0$; Update X, y, Z

```
% line search on primal
      alphap = 1;                          % initial steplength
      [dummy,posdef] = chol( X + alphap * dX );  % test if pos.def
      while posdef > 0,
              alphap = alphap * .8;
              [dummy,posdef] = chol( X + alphap * dX );
              end;
      if alphap < 1, alphap = alphap * .95; end; % stay away from boundary
% line search on dual; dZ is handled implicitly: dZ = diag( dy);
      alphad = 1;
      [dummy,posdef] = chol( Z + alphad * diag(dy) );
      while posdef > 0;
              alphad = alphad * .8;
              [dummy,posdef] = chol( Z + alphad * diag(dy) );
              end;
      if alphad < 1, alphad = alphad * .95; end;
% update
      X = X + alphap * dX;
      y = y + alphad * dy;
      Z = Z + alphad * diag(dy);
      mu = X( :)' * Z( :) / (2*n);
      if alphap + alphad > 1.8, mu = mu/2; end; % speed up for long steps
      phi = b' * y; psi = L( :)' * X( :);
% display current iteration
        disp([ iter alphap alphad (phi-psi) psi  phi ]);
```

Success of SDP Relaxation of MC

Goemans-Williamson .878 approx. algor. for MC

MC is one of Karp's NP-complete problems (APX-hard);
G-W '94 showed (with nonnegative weights on edges):

$$.87856(\text{bnd}_{SDP}) \leq \text{optvalue}_{MC} \leq \text{bnd}_{SDP}$$

Extensions/Numerics

This result has been extended (e.g. Nesterov/97) to more general quadratic functions to obtain a $\frac{\pi}{2}$ guarantee
In practice, the strength of the bound is much tighter; large problems can be solved (many authors).

SDP arise from general quadratic approximations?

General Quadratic Approximations

Approximations from quadratic functions are stronger than from linear functions. E.g.

$x \in \{\pm 1\}$ iff $x^2 = 1$ | $x \in \{0,1\}$ iff $x^2 - x = 0$

QQPs

Let

$$q_i(y) = \frac{1}{2} y^T Q_i y + y^T b_i + c_i, \; y \in \Re^n$$

$$(QQP) \qquad \begin{array}{rll} q^* = & \min & q_0(y) \\ & \text{s.t.} & q_i(y) \leq 0 \\ & & i = 1, \ldots m \end{array}$$

Lagrangian Relaxation

Lagrangian; x Lagrange multiplier vector

$$L(y,x) \quad = \quad q_0(y) + \sum_{i=1}^m x_i q_i(y)$$

or equivalently (combine quad./lin. terms)

$$\begin{aligned} L(y,x) \quad = \quad & \tfrac{1}{2} y^T (Q_0 + \sum_{i=1}^m x_i Q_i) y \\ & + y^T (b_0 + \sum_{i=1}^m x_i b_i) \\ & + (c_0 + \sum_{i=1}^m x_i c_i) \end{aligned}$$

Weak Duality

Use hidden constraints

$$d^* = \max_{x \geq 0} \min_y L(y,x) \leq q^* = \min_y \max_{x \geq 0} L(y,x)$$

Homogenization

Homogenize the Lagrangian

multiply linear term by new variable y_0:

$y_0 y^T(b_0 + \sum_{i=1}^m x_i b_i), \quad y_0^2 = 1$

use: strong duality for TRS; hidden SDP constraints

$$
\begin{array}{rcll}
d^* & = & \max\limits_{x\geq 0} \min_y & L(y,x) \\
 & = & \max\limits_{x\geq 0} \min\limits_{y_0^2=1} & \frac{1}{2} y^T(Q_0 + \sum_{i=1}^m x_i Q_i)y + t y_0^2 \\
 & & & \quad + y_0 y^T(b_0 + \sum_{i=1}^m x_i b_i) \\
 & & & \quad\quad + (c_0 + \sum_{i=1}^m x_i c_i) - t \\
 & = & \max\limits_{x\geq 0, t} \min\limits_{y} & \frac{1}{2} y^T(Q_0 + \sum_{i=1}^m x_i Q_i)y + t y_0^2 \\
 & & & \quad + y_0 y^T(b_0 + \sum_{i=1}^m x_i b_i) \\
 & & & \quad\quad + (c_0 + \sum_{i=1}^m x_i c_i) - t
\end{array}
$$

Hidden SDP Constraint in Lagrangian Dual

Hessian is $\succeq 0$

$$B := \begin{pmatrix} 0 & b_0^T \\ b_0 & Q_0 \end{pmatrix},$$

$$\mathcal{A}\begin{pmatrix} t \\ x \end{pmatrix} := -\begin{bmatrix} t & \sum_{i=1}^m x_i b_i^T \\ \sum_{i=1}^m x_i b_i & \sum_{i=1}^m x_i Q_i \end{bmatrix}, \qquad : \mathbb{R}^{m+1} \to \mathcal{S}^{n+1}$$

and the SDP constraint

$$B - \mathcal{A}\begin{pmatrix} t \\ x \end{pmatrix} \succeq 0.$$

NOTE: There is NO hidden constraint needed in convex case; e.g. if all q_i are convex.

Lagrangian Relaxation and Equivalent SDP

Dual-Primal Programs

Lagrangian Relaxation is equivalent to SDP (with $c_0 = 0$)

$$(\textbf{DSDP}) \qquad \begin{array}{rll} d^* = & \sup & -t + \sum_{i=1}^m x_i c_i \\ & \text{s.t.} & \mathcal{A}\begin{pmatrix} t \\ x \end{pmatrix} \preceq B \\ & & x \in \Re^m, t \in \Re \end{array}$$

As in LP, Dual of Dual; Use Opt. Strategy of Competing Player

$$(\textbf{DD}) \qquad \begin{array}{rll} d^* \leq p^* := & \inf & \operatorname{trace} BY \\ & \text{s.t.} & \mathcal{A}^* Y = \begin{pmatrix} -1 \\ c \end{pmatrix} \\ & & Y \succeq 0. \end{array}$$

Quadratic Assignment Problem, (QAP)

QAP Problem

- n facilities i, l, A_{il} flow or weight;
  n locations j, k, B_{jk} distances;
  C_{ij} location costs
- $n \geq 16$ considered hard; SDP provides strong (though expensive) bounds/1998;
- Nugent $n = 30$ solved for first time using weakened SDP relaxation on computational grids (CONDOR)/2002;
- Exploit group symmetry in SDP relaxation of QAP; major advance in size and efficiency/2007

QAP Formulation

QAP Applications

designing of facility layouts; VLSI design (location of modules on chips); campus planning; scheduling; process communication; turbine balancing; typewriter keyboard design; many more . . .

QAP Trace Formulation/Model

$$\text{(QAP)} \quad \mu^* := \min_{X \in \Pi} \ \operatorname{trace} AXBX^T - 2CX^T$$

$A, B, C \in \mathcal{M}^n$; Π set of permutaion matrices.

Quadratic Assignment Problem, (QAP)

Permutation Matrices

$$\begin{aligned}\Pi &= \{n \times n : (0,1), \text{row/col sums } 1\} \\ &= \{X \in \mathcal{M}^n : X \circ X = X, Xe = X^T e = e\} \\ &= \{X \in \mathcal{M}^n : X^T X = I, Xe = X^T e = e, X \geq 0\} \\ &= \{X \in \mathcal{M}^n : X^T X = XX^T = I, Xe = X^T e = e, \\ &\qquad X \circ X = X, X \geq 0\}\end{aligned}$$

QQP Model of QAP/Add Redundant Constraints

$$(\text{QAP}_E) \qquad \begin{array}{rll} \mu^* := & \min & \operatorname{trace} AXBX^T - 2CX^T \\ & \text{s.t.} & XX^T = I, X^T X = I \\ & & Xe = X^T e = e \\ & & X_{ij}^2 - X_{ij} = 0, \ \forall i, j. \end{array}$$

Lagrangian Relaxation of QAP

Find SDP relaxation of QAP by taking dual of dual

(ignore $Xe = X^Te = e$ for now)

• Add $(0,1)$-constraints to objective function; use Lagrange multipliers W_{ij}

$$\mu_{\mathcal{O}} = \min_{\substack{XX^T=I \\ X^TX=I}} \max_{W} \operatorname{trace} AXBX^T - 2CX^T + \sum_{ij} W_{ij}(X_{ij}^2 - X_{ij})$$

homogenize obj. fn; multiply by a constrained scalar x_0

$$\mu_{\mathcal{O}} \geq \mu_R = \max_{W} \min_{\substack{XX^T=X^TX=I \\ x_0^2=1}} \operatorname{trace} \left[AXBX^T + W(X \circ X)^T - x_0(2C + W)X^T\right].$$

Lagrangian Relaxation/Dual

Grouping: quadratic, linear, constant terms

Lagrange multiplier w_0 for constraint on x_0; Lagrange multipliers S_b for $XX^T = I$, S_o for $X^TX = I$

$$\begin{aligned}\mu_{\mathcal{O}} \geq \mu_R \quad := \quad \max_{W} \min_{X,\, x_0} \quad & \operatorname{trace}\left[AXBX^T + W(X \circ X)^T + w_0 x_0^2\right. \\ & \left. + S_b XX^T + S_o X^T X\right] \\ & - \operatorname{trace} x_0(2C + W)X^T \\ & - w_0 - \operatorname{trace} S_b - \operatorname{trace} S_o.\end{aligned}$$

Lagrangian Relaxation/Dual

Vectorize X

define $x := \mathrm{vec}\, X$, $y^T := (x_0, x^T)$ and $w^T := (w_0, \mathrm{vec}\, W^T)$

$$\begin{aligned}\mu_R \;=\; \max_{w}\min_{y} \quad & y^T\left[L_Q + Arrow(w) + \mathrm{B}^0\mathrm{Diag}\,(S_b) + \mathrm{O}^0\mathrm{Diag}\,(S_o)\right] y \\ & -w_0 - \mathrm{trace}\, S_b - \mathrm{trace}\, S_o\end{aligned}$$

Linear Transformations

$$L_Q := \begin{bmatrix} 0 & -\mathrm{vec}(C)^T \\ -\mathrm{vec}(C) & B \otimes A \end{bmatrix}, \quad (n^2+1) \times (n^2+1)$$

$$\mathrm{Arrow}(w) := \begin{bmatrix} w_0 & -\frac{1}{2} w_{1:n^2}^T \\ -\frac{1}{2} w_{1:n^2} & \mathrm{Diag}(w_{1:n^2}) \end{bmatrix},$$

$$\mathrm{B}^0\mathrm{Diag}(S) := \begin{bmatrix} 0 & 0 \\ 0 & I \otimes S_b \end{bmatrix}$$

and

$$\mathrm{O}^0\mathrm{Diag}(S) := \begin{bmatrix} 0 & 0 \\ 0 & S_o \otimes I \end{bmatrix}.$$

Dual of Dual

Hidden Semidefinite Constraint Yields the Equivalend SDP

$$
(D_{\mathcal{O}}) \quad \begin{array}{ll} \max & -w_0 - \operatorname{trace} S_b - \operatorname{trace} S_o \\ \text{s.t.} & L_Q + \operatorname{Arrow}(w) + \\ & \mathrm{B}^0\operatorname{Diag}(S_b) + \mathrm{O}^0\operatorname{Diag}(S_o) \succeq 0, \end{array}
$$

dual of this dual yields the semidefinite relaxation; $Y \succeq 0$ is $(n^2+1) \times (n^2+1)$, the dual matrix variable

$$
(SDP_{\mathcal{O}}) \quad \begin{array}{llll} \min & \operatorname{trace} L_Q Y & & \\ \text{s.t.} & \mathrm{b}^0\operatorname{diag}(Y) = I & & \mathrm{o}^0\operatorname{diag}(Y) = I \\ & \operatorname{arrow}(Y) = e_0 & & Y \succeq 0 \end{array}
$$

Lagrangian Relaxation/Dual

Adjoint Operators

$$\mathrm{arrow}\,(Y) := \mathrm{diag}\,(Y) - (0, (Y_{0,1:n^2})^T.$$

$$\mathrm{b}^0\mathrm{diag}\,(Y) := \sum_{k=1}^{n} Y_{(k-1)n+1:kn,(k-1)n+1:kn}$$

$$[\mathrm{o}^0\mathrm{diag}\,(Y)]_{ij} := \mathrm{trace}\; Y_{(i-1)n+1:in,(j-1)n+1:jn}$$

Direct Approach to SDP Relaxation

Vectorize Permutation Matrix

$X \in \Pi$; $x = \mathrm{vec}\,(X)$, $c = \mathrm{vec}\,(C)$.

$$\begin{array}{rcl} q(X) & = & \operatorname{trace} AXBX^T - 2CX^T \\ & = & x^T(B \otimes A)x - 2c^Tx \\ & = & \operatorname{trace} xx^T(B \otimes A) - 2c^Tx \\ & = & \operatorname{trace} L_Q Y_X, \end{array}$$

$$Y_X := \begin{bmatrix} 1 & x^T \\ x & xx^T \end{bmatrix}$$

Loss of Slater CQ

Comments

After adding the row/column sum constraints $Xe = X^T e = e$, we get that Slater's CQ fails; but we can explicitly regularize, i.e. find the smallest face/minimal cone. the SDP relaxation provides strong bounds, but expensive. Exploit group symmetries/special structure.

Sensor Network Localization, SNL, (Krislock and W.)

Wireless Sensor Network

- n ad hoc wireless sensors (nodes) to locate in $\mathbb{R}^r$,
  (r is embedding dimension;
  sensors $p_i \in \mathbb{R}^r, i \in V := 1, \ldots, n$)
- m of the sensors are anchors, $p_i, i = n - m + 1, \ldots, n$)
  (e.g. using GPS)
- pairwise distances known within radio range R,
  $D_{ij} = \|p_i - p_j\|^2, ij \in E$

Current Techniques: Nearest, Weighted, SDP Approx.

$\min_{Y \succeq 0, Y \in \Omega} \|H \circ (\mathcal{K}(Y) - D)\|$

SDP program is: Expensive/low accuracy/implicitly highly degenerate (cliques restrict ranks of feasible Ys)

Applications

Tracking Humans/Animals/Equipment

- monitoring: natural habitat; earthquakes and volcanos; weather and ocean currents.
- military, tracking of goods, vehicle positions, surveillance, (where open-air positioning is not feasible), random deployment in inaccessible terrains or disaster relief operations

Underlying Graph Realization/Partial EDM

Graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \omega)$

- node set $\mathcal{V} = \{1, \ldots, n\}$
- edge set $(i,j) \in \mathcal{E}$; $\omega_{ij} = \|p_i - p_j\|^2$ known approximately
- The anchors form a clique (complete subgraph)
- Realization of $\mathcal{G}$ in $\Re^r$: a mapping of node $v_i \to p_i \in \Re^r$ with squared distances given by ω.

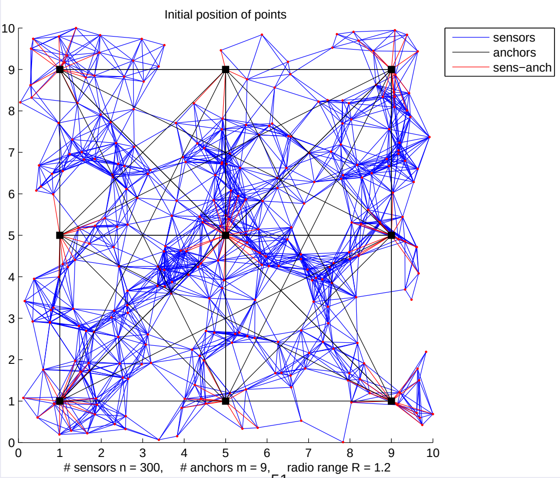
Corresponding Partial Euclidean Distance Matrix, EDM

$$D_{ij} = \begin{cases} d_{ij}^2 & \text{if } (i,j) \in \mathcal{E} \\ 0 & \text{otherwise,} \end{cases}$$

$d_{ij}^2 = \omega_{ij}$ are known squared Euclidean distances between sensors p_i, p_j; anchors correspond to a clique.

Sensor Localization Problem/Partial EDM

Sensors ○ and Anchors ■



Further Notation

Matrix with Fixed Principal Submatrix

For $Y \in \mathcal{S}^n$, $\alpha \subseteq \{1, \dots, n\}$: $Y[\alpha]$ denotes principal submatrix formed from rows & cols with indices α.

Sets with Fixed Principal Submatrices

If $|\alpha| = k$ and $\bar{Y} \in \mathcal{S}^k$, then:
$\mathcal{S}^n(\alpha, \bar{Y}) := \left\{Y \in \mathcal{S}^k : Y[\alpha] = \bar{Y}\right\}$,
$\mathcal{S}^n_+(\alpha, \bar{Y}) := \left\{Y \in \mathcal{S}^k_+ : Y[\alpha] = \bar{Y}\right\}$
i.e. the subset of matrices $Y \in \mathcal{S}^k$ ($Y \in \mathcal{S}^k_+$) with principal submatrix $Y[\alpha]$ fixed to $\bar{Y}$.

SNL Connection to SDP (Lin. Trans., Adjoints)

$v = \mathrm{diag}\,(S) \in \mathbb{R}^n$, $S = \mathrm{Diag}\,(v) \in \mathcal{S}^n$

$\mathrm{diag} = \mathrm{Diag}^*$ (adjoint)
for $B \in \mathcal{S}^n$, $\mathrm{offDiag}\,(B) := B - \mathrm{Diag}\,(\mathrm{diag}\,(B))$

$D = \mathcal{K}(B) \in \mathcal{E}^n$, $B = \mathcal{K}^\dagger(D) \in \mathcal{S}^n \cap \mathcal{S}_C$

Let $P^T = \begin{bmatrix} p_1 & p_2 & \dots & p_n \end{bmatrix} \in \mathcal{M}^{r \times n}$;
$B := PP^T \in \mathcal{S}^n$; $D \in \mathcal{E}^n$ be corresponding EDM.

$$
\begin{array}{rcl}
(\text{from } \mathcal{S}^n) \quad \mathcal{K}(B) & := & \mathcal{D}_e(B) - 2B \\
& := & \mathrm{diag}(B)\, e^T + e\, \mathrm{diag}(B)^T - 2B \\
& = & \left(p_i^T p_i + p_j^T p_j - 2 p_i^T p_j \right)_{i,j=1}^n \\
& = & \left(\| p_i - p_j \|_2^2 \right)_{i,j=1}^n \\
& = & D \quad (\text{to } \mathcal{E}^n).
\end{array}
$$

$\mathcal{K} : \mathcal{S}^n_+ \cap \mathcal{S}_C \to \mathcal{E}^n$

Linear Transformations: $\mathcal{D}_v(B), \mathcal{K}(B), \mathcal{T}(D)$

- allow: $\mathcal{D}_v(B) := \operatorname{diag}(B)\, v^T + v\, \operatorname{diag}(B)^T$;
  $\mathcal{D}_v(y) := yv^T + vy^T$
- adjoint $\mathcal{K}^*(D) = 2(\operatorname{Diag}(De) - D)$.
- $\mathcal{K}$ is $1-1$, onto between centered and hollow subspaces:
  $\mathcal{S}_C := \{B \in \mathcal{S}^n : Be = 0\}$;
  $\mathcal{S}_H := \{D \in \mathcal{S}^n : \operatorname{diag}(D) = 0\} = \mathcal{R}(\operatorname{offDiag})$
- $J := I - \frac{1}{n}ee^T$ (orthogonal projection onto $M := \{e\}^\perp$);
- $\boxed{\mathcal{T}(D) := -\frac{1}{2}J\operatorname{offDiag}(D)J \qquad (= \mathcal{K}^\dagger(D))}$

Properties of Linear Transformations

$\mathcal{K}$, $\mathcal{T}$, Diag, $\mathcal{D}_e$

$$\mathcal{R}(\mathcal{K}) = \mathcal{S}_H; \qquad \underline{\mathcal{N}(\mathcal{K}) = \mathcal{R}(\mathcal{D}_e)};$$

$$\mathcal{R}(\mathcal{K}^*) = \mathcal{R}(\mathcal{T}) = \mathcal{S}_C; \qquad \mathcal{N}(\mathcal{K}^*) = \mathcal{N}(\mathcal{T}) = \mathcal{R}(\mathrm{Diag});$$

$$\mathcal{S}^n = \mathcal{S}_H \oplus \mathcal{R}(\mathrm{Diag}) = \mathcal{S}_C \oplus \mathcal{R}(\mathcal{D}_e).$$

$\mathcal{T}(\mathcal{E}^n) = \mathcal{S}^n_+ \cap \mathcal{S}_C$ and $\mathcal{K}(\mathcal{S}^n_+ \cap \mathcal{S}_C) = \mathcal{E}^n.$

Single Clique/Facial Reduction

$\bar{D} \in \mathcal{E}^k$, $\alpha \subseteq 1{:}n$, $|\alpha| = k$

Define $\mathcal{E}^n(\alpha, \bar{D}) := \left\{ D \in \mathcal{E}^n : D[\alpha] = \bar{D} \right\}$.

Given $\bar{D}$; find a corresponding $B \succeq 0$; find the corresponding face; find the corresponding subspace.

Basic Theorem for Single Clique/Facial Reduction

THEOREM 1: Single Clique Reduction

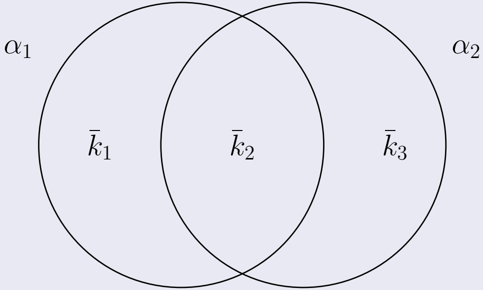
Let $\bar{D} := D[1:k] \in \mathcal{E}^k$, $k < n$, with embedding dimension $t \leq r$, and $B := \mathcal{K}^\dagger(\bar{D}) = \bar{U}_B S \bar{U}_B^T$, where $\bar{U}_B \in \mathcal{M}^{k\times t}$, $\bar{U}_B^T \bar{U}_B = I_t$, and $S \in \mathcal{S}^t_{++}$. Furthermore, let $U_B := \begin{bmatrix} \bar{U}_B & \frac{1}{\sqrt{k}} e \end{bmatrix} \in \mathcal{M}^{k\times(t+1)}$, $U := \begin{bmatrix} U_B & 0 \\ 0 & I_{n-k} \end{bmatrix}$, and let $\begin{bmatrix} V & \frac{U^T e}{\|U^T e\|} \end{bmatrix} \in \mathcal{M}^{n-k+t+1}$ be orthogonal. Then:

$$\begin{aligned} \operatorname{face} \mathcal{K}^\dagger\left(\mathcal{E}^n(1:k, \bar{D})\right) &= \left(U \mathcal{S}_+^{n-k+t+1} U^T\right) \cap \mathcal{S}_C \\ &= (UV)\mathcal{S}_+^{n-k+t}(UV)^T \end{aligned}$$

Note that we add $\frac{1}{\sqrt{k}} e$ to represent $\mathcal{N}(\mathcal{K})$; then we use V to eliminate e to recover *centered* face.

Positive Integers for Intersecting Cliques

Two Intersection Sets, α_1, α_2.



For each clique $|\alpha| = k$, we get a corresponding face/subspace ($k \times r$ matrix) representation. We now see how to handle two cliques, α_1, α_2, that intersect.

Two (Intersecting) Clique Reduction/Subsp. Repres.

THEOREM 2: Clique Intersection Reduction/Subsp. Repres.

$$\alpha_1 := 1{:}(\bar{k}_1+\bar{k}_2), \quad \alpha_2 := (\bar{k}_1+1){:}(\bar{k}_1+\bar{k}_2+\bar{k}_3) \quad \subseteq \quad 1{:}n,$$
$$k_1 := |\alpha_1| = \bar{k}_1+\bar{k}_2, \quad k_2 := |\alpha_2| = \bar{k}_2+\bar{k}_3,$$
$$k := \bar{k}_1+\bar{k}_2+\bar{k}_3.$$

For $i=1,2$, let $\bar{D}_i := D[\alpha_i] \in \mathcal{E}^{k_i}$, with embedding dimension t_i, and $B_i := \mathcal{K}^{\dagger}(\bar{D}_i) = \bar{U}_i S_i \bar{U}_i^T$, where $\bar{U}_i \in \mathcal{M}^{k_i \times t_i}$, $\bar{U}_i^T \bar{U}_i = I_{t_i}$, and $S_i \in \mathcal{S}^{t_i}_{++}$. Furthermore, for $i=1,2$, let $U_i := \begin{bmatrix} \bar{U}_i & \frac{1}{\sqrt{k_i}} e \end{bmatrix} \in \mathcal{M}^{k_i \times (t_i+1)}$, and let $\bar{U} \in \mathcal{M}^{k \times (t+1)}$ satisfy

$$\mathcal{R}(\bar{U}) = \mathcal{R}\left(\begin{bmatrix} U_1 & 0 \\ 0 & I_{\bar{k}_3} \end{bmatrix}\right) \cap \mathcal{R}\left(\begin{bmatrix} I_{\bar{k}_1} & 0 \\ 0 & U_2 \end{bmatrix}\right), \text{with } \bar{U}^T \bar{U} = I_{t+1}$$

cont...

Two (Intersecting) Clique Reduction, cont...

THEOREM 2 Nonsing. Clique Inters. cont...

cont...with

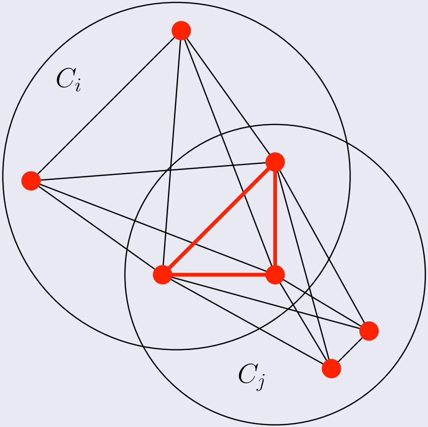
$$\mathcal{R}(\bar{U}) = \mathcal{R}\left(\begin{bmatrix} U_1 & 0 \\ 0 & I_{\bar{k}_3} \end{bmatrix}\right) \cap \mathcal{R}\left(\begin{bmatrix} I_{\bar{k}_1} & 0 \\ 0 & U_2 \end{bmatrix}\right), \text{ with } \bar{U}^T\bar{U} = I_{t+1},$$

let $U := \begin{bmatrix} \bar{U} & 0 \\ 0 & I_{n-k} \end{bmatrix} \in \mathcal{M}^{n\times(n-k+t+1)}$ and let $\begin{bmatrix} V & \frac{U^Te}{\|U^Te\|} \end{bmatrix} \in \mathcal{M}^{n-k+t+1}$ be orthogonal. Then

$$\begin{aligned} \underline{\bigcap_{i=1}^2 \operatorname{face} \mathcal{K}^\dagger\left(\mathcal{E}^n(\alpha_i, \bar{D}_i)\right)} &= \left(U\mathcal{S}_+^{n-k+t+1}U^T\right) \cap \mathcal{S}_C \\ &= (UV)\mathcal{S}_+^{n-k+t}(UV)^T \end{aligned}$$

Basic work: find $\bar{U}$ to repres. inters. of 2 subspaces.

Two (Intersecting) Clique Reduction Figure



Completion: missing distances can be recovered if desired.

Two (Intersecting) Clique Explicit Completion

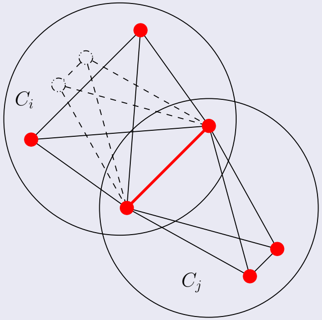
COR. Intersection with Embedding Dim. r/Completion

Hypotheses of Theorem 2 holds. Let $\bar{D}_i := D[\alpha_i] \in \mathcal{E}^{k_i}$, for $i = 1, 2$, $\beta \subseteq \alpha_1 \cap \alpha_2, \gamma := \alpha_1 \cup \alpha_2, \bar{D} := D[\beta], B := \mathcal{K}^\dagger(\bar{D}), \quad \bar{U}_\beta := \bar{U}(\beta, :)$, where $\bar{U} \in \mathcal{M}^{k \times (t+1)}$ satisfies intersection equation of Theorem 2. Let $\begin{bmatrix} \bar{V} & \frac{\bar{U}^T e}{\|\bar{U}^T e\|} \end{bmatrix} \in \mathcal{M}^{t+1}$ be orthogonal. Let $\boxed{Z := (J\bar{U}_\beta \bar{V})^\dagger B((J\bar{U}_\beta \bar{V})^\dagger)^T}$. If the embedding dimension for $\bar{D}$ is r, <u>THEN</u> $t = r$ in Theorem 2, and $Z \in \mathcal{S}^r_+$ is the unique solution of the equation $(J\bar{U}_\beta \bar{V})Z(J\bar{U}_\beta \bar{V})^T = B$, and the exact completion is

$$\boxed{D[\gamma] = \mathcal{K}\left((\bar{U}\bar{V})Z(\bar{U}\bar{V})^T\right)}.$$

2 (Inters.) Clique Reduction/Singular Case

Two (Intersecting) Clique Reduction/Singular Case



Use *R* as lower bound in singular/nonrigid case.

Algorithm

Initialize

$C_i := \left\{ j : (D_p)_{ij} < (R/2)^2 \right\}$, for $i = 1, \ldots, n$

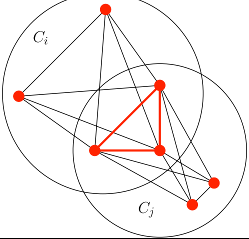
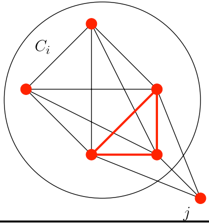
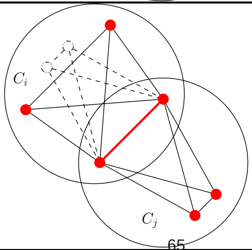
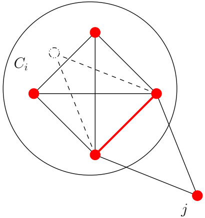
Iterate

- For $|C_i \cap C_j| \geq r + 1$, do Rigid Clique Union
- For $|C_i \cap \mathcal{N}(j)| \geq r + 1$, do Rigid Node Absorption
- For $|C_i \cap C_j| = r$, do Non-Rigid Clique Union (lower bnds)
- For $|C_i \cap \mathcal{N}(j)| = r$, do Non-Rigid Node Absorp. (lower bnds)

Finalize

When $\exists$ a clique containing all anchors, use computed facial representation and positions of anchors to solve for X

Algorithm

	Clique Union	Node Absorption
Rigid		
Non-rigid		

Results - Data for Random Noisless Problems

- 2.16 GHz Intel Core 2 Duo, 2 GB of RAM
- Dimension $r = 2$
- Square region: $[0,1] \times [0,1]$
- $m = 9$ anchors
- Using only Rigid Clique Union and Rigid Node Absorption
- Error measure: Root Mean Square Deviation

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

Results - Large n (SDP size $O(n^2)$)

n # of Sensors Located

n # sensors \ R	0.07	0.06	0.05	0.04
2000	2000	2000	1956	1374
6000	6000	6000	6000	6000
10000	10000	10000	10000	10000

CPU Seconds

# sensors \ R	0.07	0.06	0.05	0.04
2000	1	1	1	3
6000	5	5	4	4
10000	10	10	9	8

RMSD (over located sensors)

n # sensors \ R	0.07	0.06	0.05	0.04
2000	$4e-16$	$5e-16$	$6e-16$	$3e-16$
6000	$4e-16$	$4e-16$	$3e-16$	$3e-16$
10000	$3e-16$	$5e-16$	$4e-16$	$4e-16$

Results - N Huge SDPs Solved

Large-Scale Problems

# sensors	# anchors	radio range	RMSD	Time
20000	9	.025	$5e{-16}$	25s
40000	9	.02	$8e{-16}$	1m 23s
60000	9	.015	$5e{-16}$	3m 13s
100000	9	.01	$6e{-16}$	9m 8s

Size of SDPs Solved: $N = \binom{n}{2}$ (# vrbls)

$\mathcal{E}_n(\text{density of } \mathcal{G}) = \pi R^2$; $M = \mathcal{E}_n(|E|) = \pi R^2 N$ (# constraints)

Size of SDP Problems:

$M = [3,078,915 \quad 12,315,351 \quad 27,709,309 \quad 76,969,790]$

$N = 10^9\, [0.2000 \quad 0.8000 \quad 1.8000 \quad 5.0000]$

Summary

- Many hard (combinatorial) problems can be modelled using quadratic objectives and constraints, QQPs.
- QQPs are generally NP-hard problems. But, the Lagrangian relaxation can be solved efficiently using the equivalent SDP (relaxation).
- The special structure of the SDP relaxations can be exploited in order to get efficient solutions for large scale problems.
- Many SDP relaxations of combinatorial problems are degenerate. But, this degeneracy can be exploited. In particular, the SDP relaxation of SNL is highly (implicitly) degenerate. This degeneracy allows for a fast, accurate solution technique.

Thanks for your attention!

Semidefinite Programming
Applications and Implementations

Henry Wolkowicz

Dept. of Combinatorics and Optimization
University of Waterloo

CORS/MOPGP
Niagara Falls, June 11, 2012