Newton Directions and Symmetrization
in
Primal-Dual Interior-Point Methods
for
Semidefinite Programming

Henry Wolkowicz
University of Waterloo
(with Serge Kruk and Franz Rendl)

Outline

What is semidefinite programming, SDP (quick)

Complementarity Conditions
and Symmetrization

Ill-conditioning

Search directions

Lots of questions
(from a numerical analysis point of view)

Semidefinite Programming
looks just like
Linear Programming

$$\textbf{(PSDP)} \qquad \begin{array}{rlll} p^* = & \text{max} & \text{trace}\, CX & (\langle C, X \rangle) \\ & \text{s.t.} & \mathcal{A}X = b & \text{linear constr.} \\ & & X \succeq 0, \quad (X \in \mathcal{P}) & \text{nonneg constr.} \end{array}$$

$\preceq$ denotes the Löwner partial order
$A \preceq B$ if $B - A \succeq 0$

$\mathcal{S}_n$ denotes $n \times n$ symmetric matrices

$$\mathcal{A} : \mathcal{S}_n \to \Re^m$$

$$(\mathcal{A}X)_i = \text{trace}\,(A_i X), \text{ for given } A_i \in \mathcal{S}_n$$

$\mathcal{P}$ - cone of positive semidefinite matrices

replaces

$\Re^n_+$ - nonnegative orthant

Why use SEMIDEFINITE PROGRAMMING?

Quadratic approximations are better than linear approximations.

Quadratic approximations are too hard to solve!

But, we can solve relaxations of quadratic approximations efficiently using semidefinite programming.

APPLICATIONS

Finding bounds and good feasible solutions for **hard combinatorial problems** such as: max-cut; graph partitioning; quadratic assignment problem; max-clique.

Unconstrained and constrained **NLP** techniques, e.g.: quasi-Newton updates that preserve positive definiteness; Trust region algorithms for large scale minimization; Extended SQP techniques for constrained minimization.

Partial Hermitian **matrix completion problems**.

Engineering problems such as: Ricatti equations; min-max eigenvalue problems; matrix norm minimization; eigenvalue localization.

SIMILARITIES WITH LP

DUALITY

payoff function, player Y to player X (Lagrangian)

$$L(X,y) := \text{trace}\,(CX) + y^t(b - \mathcal{A}X)$$

Optimal (worst case) strategy for player X:

$$p^* = \max_{X \succeq 0} \min_{y} L(X,y)$$

Using the *hidden constraint* $b - \mathcal{A}X = 0$, recovers primal problem.

$$
\begin{aligned}
L(X,y) &= \text{trace}\,(CX) + y^t(b - \mathcal{A}X) \\
&= b^t y + \text{trace}\;(C - \mathcal{A}^* y)\,X
\end{aligned}
$$

adjoint operator, $\mathcal{A}^* y = \sum_i y_i A_i$

$$\langle \mathcal{A}^* y, X \rangle = \langle y, \mathcal{A}X \rangle\,, \quad \forall X, y$$

$$p^* = \max_{X \succeq 0} \min_{y} L(X,y) \leq d^* := \min_{y} \max_{X \succeq 0} L(X,y)$$

The dual is obtained from the optimal strategy of the competing player, Y.
The *hidden constraint* $C - \mathcal{A}^* y \preceq 0$
yields the dual

$$(\mathbf{DSDP}) \quad \begin{array}{rll} d^* = & \min & b^t y \\ & \text{s.t.} & \mathcal{A}^* y \succeq C \end{array}$$

for the primal

$$(\mathbf{PSDP}) \quad \begin{array}{rll} p^* = & \max & \operatorname{trace} CX \\ & \text{s.t.} & \mathcal{A} X = b \\ & & X \succeq 0 \end{array}$$

Characterization of optimality for the
dual pair X, y (slack $Z \succeq 0$)

$\mathcal{A}^* y - Z = C$ dual feasibility

$AX = b$ primal feasibility

$ZX = 0$ complementary slackness

$ZX = \mu I$ perturbed

Forms the basis for:

primal simplex method
dual simplex method
interior point methods

Other dual pairs:

$$\textbf{(P)}\quad p^* = \begin{array}{cc} \max & b^t y \\ \text{s.t.} & \mathcal{A}^* y \preceq C \end{array}$$

$$\textbf{(D)}\quad d^* = \begin{array}{cc} \min & \operatorname{trace} CX \\ \text{s.t.} & \mathcal{A}X = b \\ & X \succeq 0 \end{array}$$

However, there are several subtle
DIFFERENCES with LP.

1. Duality gaps can exist for SDP in the absence of strictly feasible solutions (Slater's constraint qualification, CQ).

2. Strict complementarity can fail at the optimum.

$\Rightarrow$ 3. The perturbed optimality conditions involve nondiagonal matrices and map between different spaces
(overdetermined -requires symmetrization).

1. Duality gaps:

Example

If the primal is

$$
\textbf{(P)} \qquad
\begin{array}{rll}
p^* = & \max & x_2 \\
& \text{s. t.} & \begin{bmatrix} x_2 & 0 & 0 \\ 0 & x_1 & x_2 \\ 0 & x_2 & 0 \end{bmatrix} \preceq \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}
\end{array}
$$

then the dual is

$$
\textbf{(D)} \qquad
\begin{array}{rlc}
d^* = & \min & \operatorname{trace} U_{11} \\
& \text{s. t.} & U_{22} = 0 \\
& & U_{11} + 2U_{23} = 1 \\
& & U \succeq 0.
\end{array}
$$

Then $p^* = 0 < d^* = 1$.

2. Strict complementarity:

Example

If the primal is

$$
\begin{array}{llc}
p^* = & \max & x_1 \\
\mathbf{(P)} & \text{s. t.} & \begin{bmatrix} x_1 & x_3 & x_2 \\ x_3 & x_2 & 0 \\ x_2 & 0 & x_3 \end{bmatrix} \preceq \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}
\end{array}
$$

then the dual is

$$
\begin{array}{rlc}
d^* = & \min & \operatorname{trace} U_{33} \\
 & \text{s. t.} & U_{11} = 1 \\
\mathbf{(D)} & & U_{22} + 2U_{13} = 0 \\
 & & U_{33} + 2U_{12} = 0 \\
 & & U \succeq 0.
\end{array}
$$

Then the (unique) optimum pair is:
$U = \text{Diag}\,(1, 0, 0)$
and $x = [0, 0, 0]$ with slack $Z = \text{Diag}\,(0, 0, 1)$

INTERIOR-POINT METHODS

started with Karmarkar 1984

primal-dual interior-point methods:

1. potential reduction
2. path following
3. infeasible-interior-point
4. predictor-corrector

(see e.g. recent book by Steve Wright, or interior-point archives)

Connection between Karmarkar's algorithm and log-barrier problems (Stanford gang of 4: Gill, Murray, M. Wright, Saunders)

$$
\begin{array}{rccc}
 & d^* = & \min & b^t y \\
(\mathbf{D}) & & \text{s.t.} & \mathcal{A}^* y - Z = C \\
 & & & Z \succeq 0
\end{array}
$$

log-barrier problem for dual SDP, with fixed $\mu > 0$

$$
\begin{array}{ll}
\min & b^t y - \mu \log\det Z \\
\text{s.t.} & \mathcal{A}^* y - Z = C
\end{array}
$$

and $\mu \downarrow 0$

Lagrangian for log-barrier

$$
\min b^t y + \operatorname{trace} X \left(C - \mathcal{A}^* y + Z\right) - \mu \log\det Z
$$

Optimality conditions for the
dual log-barrier
solve $F_\mu(X, y, Z) = 0$, with $X, Z \succ 0$

$$\frac{\partial}{\partial X}: \quad C - \mathcal{A}^* y + Z = 0 \qquad \text{dual feasibility}$$

$$\frac{\partial}{\partial y}: \quad b - \mathcal{A} X = 0 \qquad \text{primal feasibility}$$

$$\frac{\partial}{\partial Z}: \quad X - \mu Z^{-1} = 0 \qquad \text{complementary slackness}$$

$ZX - \mu I = 0$ perturbed complementary slackness

Note:
the nonlinear system $F_\mu(X, y, Z) = 0$ is overdetermined if (more linear) $ZX - \mu I = 0$ is used.
(not the case for LP)

CENTRAL PATH

For each $\mu > 0$, there exists a unique (X_μ, y_μ, Z_μ), with $X_\mu, Z_\mu \succ 0$, that solves the nonlinear system $F_\mu(X, y, Z) = 0$.

$$\mathcal{C} = \{(X_\mu, y_\mu, Z_\mu) : \mu > 0\}$$

well defined,
converges to optimum as $\mu \downarrow 0$,
long steps possible in neighbourhoods near $\mathcal{C}$

p-d i-p methods followed:
Megiddo 1987, Kojima, Mizuono, and Yoshise 1987, many others ....

```
framework p-d i-p:
Given $(X^0, y^0, Z^0) \in \mathcal{F}^0$
for $k = 0, 1, 2 \ldots$
solve for the search direction

    $F'(X^k, y^k, Z^k) \begin{pmatrix} \Delta X^k \\ \Delta y^k \\ \Delta Z^k \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ -X^k Z^k + \sigma_k \mu_k I \end{pmatrix}$
    where $\sigma_k$ centering, $\mu_k = \text{trace}\, X^k Z^k / n$

$(X^{k+1}, y^{k+1}, Z^{k+1}) = (X^k, y^k, Z^k) + \alpha_k (\Delta X^k, \Delta y^k, \Delta Z^k)$
    so that $(X^{k+1}, Z^{k+1}) \succ 0$
end (for).
```

Complications:

Overdetermined system, i.e. how to symmetrize ZX, e.g. $ZX + XZ$, many others ...

line search

strict complementarity may fail

duality gaps

SYMMETRIZATION

from log-barrier we get:
$X - \mu Z^{-1} = 0$
with linearization:
$X + \Delta X - \mu \left(Z^{-1} - Z^{-1} \Delta Z Z^{-1} \right) = 0$

more linear version - multiply by Z:
$ZX - \mu I = 0$
with linearization:
$ZX - \mu I + Z \Delta X + \Delta Z X = 0$

symmetrize:
$ZX + ZX - 2\mu I = 0$
with linearization:
$ZX + ZX - 2\mu I + Z\Delta X + \Delta X Z + \Delta Z X + X \Delta Z = 0$

Others:
$Z^{1/2} X Z^{1/2} - \mu I = 0$, etc...
$H_P(M) := \frac{1}{2} \left(PMP^{-1} + (PMP^{-1})^t \right)$
$M = XZ$ and P any nonsingular matrix

$XZ + ZX$:
advantage: fast convergence and stability;
disadvantage: solve Lyapunov equation when finding Newton direction

Heuristic; symmetrize ΔX after it is calculated:
advantages: fast convergence and inexpensive Newton direction, especially for max-cut problem!
disadvantage: stability

Still many open questions, e.g. why symmetrize?
Can apply Gauss-Newton method to solving overdetermined system of nonlinear equations.

```
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);

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. de
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
psi = L(:)' * X( :);            % and primal costs
mu = Z( :)' * X( :)/( 2*n);     % initial complementarity
iter=0;                         % iteration count

disp(['      iter    alphap    alphad      gap     lower      uppe

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
      dX = ( dX + dX')/2;       % symmetrize
```

```
% line search on primal
      alphap = 1;               % initial steplength
      [dummy,posdef] = chol( X + alphap * dX );  % test if pos.de
      while posdef > 0,
              alphap = alphap * .8;
              [dummy,posdef] = chol( X + alphap * dX );
              end;
      if alphap < 1, alphap = alphap * .95; end; % stay away from
% 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 lo
      phi = b' * y; psi = L( :)' * X( :);
% display current iteration
        disp([ iter alphap alphad (phi-psi) psi  phi ]);

        end;            % end of main loop
```

486 66 Mhz, PC using a C encoded version;
# of iterations does not seem to depend significantly on n
hh gives the number of hours, mm the number of minutes, ss the seconds.
random unweighted graphs with edge probability $p = 0.5$.
stopping condition to 6 digits of accuracy

size n	iterations	hh:mm:ss
100	14	48
150	12	2:30
200	12	6:30
250	13	14:24
300	14	30:24
400	14	1:24:36
500	14	2:13.18

included triangle constraints;
favorable to use Mehrotra's predictor – corrector approach
system is solved twice with the same factorization but with different right hand sides.
The triangle inequalities were included successively according to their amount of violation
programs have n equality and $4\binom{n}{3}$ inequality constraints in the dual program.
The final solution satisfies all these constraints.
# of iterations astonishingly small
e.g. problems with $n = 100$ satisfies $4\binom{100}{3} = 646,800$ inequality constraints.

random graphs, edge-weights $\in \{-10, \ldots, 10\}$				
size n	cut	upper bnd (%)	hh:mm:ss	iters
30	330	330.0001 (0.0)	54	59
40	619	619.0000 (0.0)	2:02	60
50	708	708.0001 (0.0)	1:04:22	166
70	1215	1248.7142 (2.8)	4:42:21	220
100	2440	2540.2528 (4.1)	37:56:05	266

random unweighted graphs, edge probability 0.5				
size n	cut	upper bnd (%)	hh:mm:ss	iters
30	143	143.0000 (0.0)	56	62
40	245	245.6864 (0.3)	59:19	247
50	374	375.3689 (0.4)	1:54:18	250
70	723	726.0399 (0.4)	5:57:22	257

Simple convex program:

$$\textbf{(CP)}\ p^* = \begin{array}{ll} \text{min} & f(x) \\ \text{s.t.} & x \in \Omega \end{array}$$

where:

$f : X \to \Re$ - differentiable, convex function

$\Omega \subset X$ is an abstract inner-product space

Ω - convex set

Pshenichnyi Condition:

$x^* \in \Omega$ solves (CP) **iff** $\nabla f(x^*) \in (\Omega - x^*)^+$

polar cone

$$K^+ = \{\phi \in X^* : \langle \phi, k \rangle \geq 0,\ \forall k \in K\}$$

special case when Ω closed convex cone

$x^* \in \Omega$ solves (CP)

iff

$\nabla f(x^*) \in (\Omega)^+$ and

$\langle \nabla f(x^*), x^* \rangle = 0$ (compl. slack.)

special case when $\Omega = \Re^n_+$
if $z, x \in \Re^n_+$
$z^t x = 0$ iff $z_i x_i = 0$ for all i

special case when $\Omega = \mathcal{P}$, psd matrices
if $Z, X \in \mathcal{P}$
$\langle Z, X \rangle =$ trace $ZX = 0$ iff $ZX = 0$

However, these special cases do not hold in general. The purpose of *complicating* the complementarity condition is to obtain the *powerful* central path concept. (Avoids convergence to nonfeasible -but stationary - points.)

for some y and X feasible for $\mathcal{A}X = b$

$$p^* = \max \operatorname{trace}(CX) + y^t(b - \mathcal{A}X) \quad \text{s.t.} \quad X \succeq 0$$

Optimality Conditions,
$F(X, y, Z) = 0$, $X, Z \succeq 0$:

$$\nabla F : \begin{cases} C - \mathcal{A}^* y - Z = 0, & \text{dual feasibility} \\ \quad b - Ax = 0 & \text{primal feasibility} \\ \quad \langle Z, X \rangle = 0 & \text{complementary slackness} \end{cases}$$

Recall:
$\langle Z, X \rangle = \operatorname{trace} ZX = 0$ iff $ZX = 0$

The above system is **underdetermined**
Why not use simpler underdetermined system?

STABILITY QUESTIONS

Let $f : \Re^n \to \Re$ and $g : \Re^n \to \Re$ be twice continuously differentiable functions

(e.g. f is linear part of log-barrier problem, i.e. it is the Lagrangian of (PSDP); while $g(x) = \det(x)$, is the determinant.)

consider

$$(NLP) \qquad \begin{array}{ll} \text{min} & f(x) \\ \text{subject to} & g(x) \geq 0, \end{array}$$

Optimality under CQ:

$$\begin{array}{rcl} \nabla f(x) - \lambda \nabla g(x) & = & 0 \\ \langle \lambda, g(x) \rangle & = & 0 \\ g(x) & \geq & 0 \\ \lambda & \geq & 0. \end{array}$$

$\mu > 0$ fixed log-barrier parameter
log-barrier problem is:

$$\min_x B_\mu(x) := f(x) - \mu \log g(x).$$

stationarity

$$0 = \nabla B_\mu(x_\mu) = \nabla f(x_\mu) - \frac{\mu}{g(x_\mu)} \nabla g(x_\mu), \; g(x_\mu) > 0.$$

we can assume $x_\mu \to x^*$ and $g(x_\mu) \to 0$
Therefore

$$\frac{\mu}{g(x_\mu)} \to \lambda^*$$

Hessian

$$\nabla^2 P_\mu(x) =$$

$$\left\{\nabla^2 f(x) - \frac{\mu}{g(x)}\nabla^2 g(x)\right\} + \left[\left(\frac{\mu}{g(x)}\right)\frac{1}{g(x)}\nabla g(x)\nabla g(x)^t\right].$$

first { } part converges
second [ ] part is rank one ⇒ ill-conditioning

Fix: add new variable $\Delta v \in \Re$

$$\Delta v = -\frac{\mu}{g^2(x)}\nabla g(x)^t \Delta x.$$

Newton direction: $d = (\Delta x, \Delta v)^t$,

$$\begin{bmatrix} \nabla^2 f(x_\mu) - \frac{\mu}{g(x_\mu)}\nabla^2 g(x_\mu) & \nabla g(x_\mu) \\ \nabla g(x_\mu)^t & \frac{g(x_\mu)^2}{\mu} \end{bmatrix} \begin{pmatrix} \Delta x \\ \Delta v \end{pmatrix}$$

$$= \begin{pmatrix} -\nabla f(x_\mu) + \frac{\mu}{g(x_\mu)}\nabla g(x_\mu) \\ 0 \end{pmatrix}.$$

As $\mu \to 0$ the Hessian becomes

$$\begin{bmatrix} \nabla^2 f(x^*) - \lambda^* \nabla^2 g(x^*) & \nabla g(x^*) \\ \nabla g(x^*) & 0 \end{bmatrix}.$$

This matrix is nonsingular under the standard second order sufficient optimality conditions. i.e. that the Hessian of the Lagrangian is positive definite on the tangent space; equivalently, $\nabla^2 f(x^*) - \lambda^* \nabla^2 g(x^*)$ is positive definite on the orthogonal complement of $\nabla g(x^*)$.

alternative to overcome ill-conditioning

use perturbed KKT conditions
use substitution $\lambda = \frac{\mu}{g(x)}$ or $\lambda g(x) = \mu$

Then becomes

$$\begin{array}{rcl} \nabla f(x) - \lambda \nabla g(x) & = & 0 \\ \lambda g(x) & = & \mu \\ g(x) & > & 0 \end{array}$$

i.e. $\lambda g(x) = \mu$ avoids ill-conditioning that arises when differentiating $\lambda = \frac{\mu}{g(x)}$.

Summary:
two ways to avoid ill-conditioning of log-barrier problem.

1. increase the dimension of the problem by one - still finds the Newton direction

2. change the optimality conditions (perturb) direction based solely on the optimality conditions and not on a log-barrier problem
(not a Newton direction)

Alternate motivation for changing true log-barrier direction:

$X - \mu Z^{-1} = 0$

with linearization:

$X + \Delta X - \mu \left(Z^{-1} - Z^{-1} \Delta Z Z^{-1} \right) = 0$

is to get more linear version - multiply by Z:

$ZX - \mu I = 0$

with linearization:

$ZX - \mu I + Z \Delta X + \Delta Z X = 0$

MORE LINEARITY
SPEEDS UP NEWTON'S METHOD
but
results in overdetermined system