Semidefinite Optimization Algorithms - Basics

Franz Rendl

http://www.math.uni-klu.ac.at

Alpen-Adria-Universität Klagenfurt

Austria

Overview

- Part 1:
  A short review of conic duality
- Part 2:
  Central path and interior-point methods

Conic duality

K closed convex cone in $I\!R^n$. Data:
$c \in I\!R^n, b \in I\!R^m, A \ldots m \times n$ matrix

Linear Program over K:

$$M := \inf\{c^T x : Ax = b, x \in K\}$$

dual cone $K^* := \{y \in I\!R^n : \langle x, y\rangle \geq \forall x \in K\}$

ordinary LP, second-order cone LP, semidefinite Programming

SDP: $\inf\langle C, X\rangle : \ A(X) = b, \ X \succeq 0$

$A(X)$ is linear operator, given through $A(X)_i = \langle A_i, X\rangle$

Examples

$$M = \inf\{x_{11} : X = \begin{pmatrix} x_{11} & 1 \\ 1 & 1 \end{pmatrix} \succeq 0\}.$$

In this case $M = 1$ and $\inf$ is attained.

$$M = \inf\{x_{11} : X = \begin{pmatrix} x_{11} & 1 \\ 1 & x_{22} \end{pmatrix} \succeq 0\}.$$

Here $M = 0$, $\inf$ not attained.

Examples

$$M = \inf\{x_{11} : X = \begin{pmatrix} x_{11} & 1 \\ 1 & 1 \end{pmatrix} \succeq 0\}.$$

In this case $M = 1$ and $\inf$ is attained.

$$M = \inf\{x_{11} : X = \begin{pmatrix} x_{11} & 1 \\ 1 & x_{22} \end{pmatrix} \succeq 0\}.$$

Here $M = 0$, $\inf$ not attained.

$$M = \inf\{2x_{12} : X = \begin{pmatrix} 0 & 1 \\ 1 & x_{22} \end{pmatrix} \succeq 0\}.$$

This problem is infeasible, $M = +\infty$.

Dual Problem

Use Lagrange dual and Minimax Inequality to get Weak duality :

$$M = \inf_{Ax=b,\ x \in K} c^T x = \inf_{x \in K} \sup_{y} c^T x + y^T (b - Ax)$$

Dual Problem

Use Lagrange dual and Minimax Inequality to get Weak duality :

$$M = \inf_{Ax=b,\ x\in K} c^T x = \inf_{x\in K} \sup_{y} c^T x + y^T(b - Ax)$$

$$\geq \sup_{y} \inf_{x\in K} b^T y + (c - A^T y)^T x$$

$$= \sup_{c - A^T y \in K^*} b^T y =: M'.$$

Dual Problem

Use Lagrange dual and Minimax Inequality to get Weak duality :

$$M = \inf_{Ax=b,\ x\in K} c^T x = \inf_{x\in K} \sup_y \ c^T x + y^T(b - Ax)$$

$$\geq \sup_y \inf_{x\in K} \ b^T y + (c - A^T y)^T x$$

$$= \sup_{c-A^T y\in K^*} b^T y =: M'.$$

In general, $\sup$ and $\inf$ need not be attained, there can be strict inequality after exchanging $\sup$ and $\inf$ and also a finite (nonzero) duality gap between primal and dual value.

Asymptotically feasible solutions

The sequence (x_k) is primal asymptotically feasible, if

$$(x_k) \subseteq K, \ \lim_k Ax_k = b.$$

$$m := \inf_{(x_k) p.a.f.} \liminf_k \ c^T x_k$$

The sequence (y_k) is dual asymptotically feasible if there exists a sequence $(w_k) \subseteq K^*$ such that

$$\lim_k \ A^T y_k + w_k = c.$$

$$m' := \sup_{(y_k) d.a.f} \limsup_k \ b^T y_k.$$

Duality Theorems

by definition: $m \leq M, \; M' \leq m'$

weak duality: if x primal feasible and (y_k) dual asymptotically feasible, then $m' \leq M$

strong duality: If M or m' finite, then $M = m'$.

x is strictly feasible if $Ax = b, \; x \in int(K)$.

Attainment: If the dual has a strictly feasible solution and M' is finite, then $M = m = M' = m'$ and the primal infimum is attained.

An example

$$\inf\{x_{11} : x_{22} = 0,\ x_{11} + 2x_{23} = 1, X \succeq 0.\}$$

In this case

$$X = \begin{pmatrix} x_{11} & x_{12} & x_{13} \\ x_{12} & 0 & \frac{1}{2}(1 - x_{11}) \\ x_{13} & \frac{1}{2}(1 - x_{11}) & x_{33} \end{pmatrix} \succeq 0,$$

implies $x_{12} = 0, x_{11} = 1$, hence $M = 1$. Taking

$$X_k = \begin{pmatrix} \frac{1}{k} & 0 & 0 \\ 0 & \frac{1}{k} & \frac{1}{2}(1 - \frac{1}{k}) \\ 0 & \frac{1}{2}(1 - \frac{1}{k}) & k \end{pmatrix} \succeq 0,$$

and $A(X_k) \rightarrow b$. Note also that $(X_k)_{11} = \frac{1}{k} \rightarrow 0 \geq m$.

Back to Semidefinite Programs

primal problem:

$$(P) \quad \max\{\langle C, X\rangle : A(X) = b, X \succeq 0\}$$

dual problem:

$$(D) \quad \min\{b^T y : A^T(y) - C = Z \succeq 0\}$$

Strong duality (primal=dual and optima are attained) holds if we assume that both the primal and the dual problem have strictly feasible points ($X, Z \succ 0$).
Then (X, y, Z) is optimal if and only if

$$A(X) = b,\ X \succeq 0,\ \ A^T(y) - Z = C,\ Z \succeq 0,\ \ \langle X, Z\rangle = 0.$$

We have $m + \binom{n+1}{2} + 1$ equations, and $m + 2\binom{n+1}{2}$ variables.

Optimality conditions

$X,\ Z \succeq 0$ means $X = UU^T,\ Z = VV^T$, so we conclude that $0 = \langle X, Z \rangle = \|U^T V\|^2$ implies

$$ZX = UU^T VV^T = 0.$$

Therefore (X, y, Z) is optimal if and only if

$$A(X) = b,\ X \succeq 0,\ \ A^T(y) - Z = C,\ Z \succeq 0,\ \ ZX = 0.$$

Now $m + \binom{n+1}{2} + n^2$ equations, and $m + 2\binom{n+1}{2}$ variables. Too many equations as ZX need not be symmetric.

Central Path

We assume:

$$(A) \quad \exists \text{ primal and dual feasible points } X, Z \succ 0.$$

Consider, for $\mu > 0$ the system:

$$(CP) \quad A(X) = b, \; Z = A^T y - C, \; ZX = \mu I$$

over $X, Z \succeq 0$.
Fundamental Theorem for Interior-Point methods:

$$(CP) \text{ has unique solution } \forall \mu > 0 \iff (A) \text{ holds.}$$

This solution $(X(\mu)), y(\mu), Z(\mu))$ forms smooth curve, called Central Path.
Basic idea: follow this path until $\mu \approx 0$.

Central Path Equations

The system defining (CP) is overdetermined. Several ways to fix this:
Replace $ZX - \mu I = 0$ by

1. $Z - \mu X^{-1} = 0$
2. $X - \mu Z^{-1} = 0$
3. $ZX + XZ - 2\mu I = 0$
4. $P(.)P^{-1} + (P(.)P^{-1})^T$ Monteiro-Zhang family

These lead to different linearizations.
Path following methods: Follow the central path by finding points (close to it) for a decreasing sequence of μ.

Interior-Point Methods to solve SDP (1)

Primal-Dual Path-following Methods:

maintain $X, Z \succeq 0$ and try to reach feasibility and optimality.
Use Newton's method applied to perturbed problem
$ZX = \mu I$ or variant from before, and iterate for $\mu \to 0$.
At start of iteration: $(X \succ 0, y, Z \succ 0)$
Linearized system (CP) to be solved for $(\Delta X, \Delta y, \Delta Z)$:

$$A(\Delta X) = r_P := b - A(X) \qquad \text{primal residue}$$

$$A^T(\Delta y) - \Delta Z = r_D := Z + C - A^T(y) \qquad \text{dual residue}$$

$$Z\Delta X + \Delta Z X = \mu I - ZX \qquad \text{path residue}$$

The last equation can be reformulated in many ways, which all are derived from the complementarity condition $ZX = 0$

Interior-Point Methods to solve SDP (2)

Direct approach with partial elimination:
Using the second and third equation to eliminate ΔX and ΔZ, and substituting into the first gives

$$\Delta Z = A^T(\Delta y) - r_D, \quad \Delta X = \mu Z^{-1} - X - Z^{-1}\Delta Z X,$$

and the final system to be solved:

$$A(Z^{-1}A^T(\Delta y)X) = \mu A(Z^{-1}) - b + A(Z^{-1}r_D X)$$

Note that

$$A(Z^{-1}A^T(\Delta y)X) = M\Delta y,$$

but the $m \times m$ matrix M may be expensive to form.

Computational effort

- explicitely determine Z^{-1} $O(n^3)$
- several matrix multiplications $O(n^3)$
- final system of order m to compute Δy $O(m^3)$
- forming the final system matrix $O(mn^3 + m^2n^2)$

  recall $m_{ij} = \text{tr}(A_i Z^{-1} A_j X)$
- line search to determine

  $X^+ := X + t\Delta X, Z^+ := Z + t\Delta Z$ is at least $O(n^3)$

Effort to form system matrix M depends on structure of $A(.)$
Limitations: $n \approx 1000,\ m \approx 5000$.

Timings: Random SDP

Each A_i is nonzero only on randomly chosen 4×4 submatrix, main diagonal is 0.
SEDUMI seconds with default setting.

n	m	secs.
100	1000	11
100	2000	159
200	2000	151
200	5000	2607
300	5000	2395

No attempt with larger m. Memory (!!) and time (!!)

For more results, see Mittelmann's site:
`http://plato.asu.edu/ftp/sdplib.html`

Exploit Structure

SDP relaxation for Max-Cut:

$$\max\langle L, X\rangle : \; diag(X) = e, \; X \succeq 0.$$

Here $\langle A_i, X\rangle = e_i^T X e_i = x_{ii}$.
Therefore the system matrix $M = (m_{ij})$ has

$$m_{ij} = \text{tr} A_i Z^{-1} A_j X = e_i^T Z^{-1} e_j e_j^T X e_i = Z_{ij}^{-1} \cdot X_{ij},$$

therefore $M = Z^{-1} \circ X$. Can be formed in $O(n^2)$ instead of $O(n^4)$ steps.

Basic SDP Relaxation of Max-Cut

We solve $\max\langle L, X\rangle : \quad diag(X) = e, \quad X \succeq 0$.
Matrices of order n, and n simple equations $x_{ii} = 1$

n	seconds
1000	12
2000	102
3000	340
4000	782
5000	1570

Seconds on a PC. Implementation of primal-dual interior-point method in MATLAB, 30 lines of source code

Representation of linear equations

Given a graph $G = (V, E)$ with $|V| = n, \ |E| = m$. Notation: We write $A_G(X) = 0$ for $x_{ij} = 0, \ (ij) \in E(G)$. Hence $A_G(X)_{ij} = \langle E_{ij}, X \rangle$ with $E_{ij} = e_i e_j^T + e_j e_i^T$.

Any symmetric matrix M can therefore be written as

$$M = \text{Diag}(m) + A_G(u) + A_{\bar{G}}(v).$$

Recall theta function

$$\vartheta(G) = \max\{\langle J, X \rangle : \text{tr}(X) = 1, A_G(X) = 0, \ X \succeq 0\}$$

$$= \min\{t : tI + A_G^T(y) - J \succeq 0\}.$$

The number of equations depends on the edge set E.

Theta for sparse and dense graphs

For dense graphs, we can use the following reformulation. Let $Y = tI + A_G^T(y)$ and set $Z = Y - J$ which has the following properties:

$A_{\bar{G}}(Z) = -2e$, because $z_{ij} = -1$ for $[ij] \notin E$.

$te -$ diag$(Z) = e$, because diag$(Y) = te$. Hence we get the theta function equivalently as

$$\vartheta(G) = \min\{t : te - \textbf{diag}(Z) = e,\ -A_{\bar{G}} = 2e, Z \succeq 0\} =$$

$$\max\{e^T x + 2e^T \xi : \textbf{Diag}(x) + A_{\bar{G}}(\xi) \succeq 0, e^T x = 1\}.$$

Here the dual has $\bar{m} + n$ equations, hence this is good for dense graphs ($\bar{m}$ small in this case).

Comparing the two models

The two models have the following running times on graphs with $n = 100$ and various edge densities.

density	0.90	0.75	0.50	0.25	0.10
m	4455	3713	2475	1238	495
dense	1	7	42	130	238
sparse	223	118	34	5	1

Comparison of the computation times (in seconds) for ϑ on five random graphs with 100 vertices and different densities in the dense and the sparse model.

Theta Function -limits of Interior Points

Sparse model, $m \leq \frac{1}{4}n^2$ The system to be solved is of size $|E|$.

n	100	200	300	400
$\|E\|$	487	2047	4531	7949
time	1	30	309	1583
$\|E\|$	1240	5099		
time	7	371		
$\|E\|$	2531	10026		
time	34	2735		

Impractical, once system size is of order 10^4.

What if m is too large?

We consider

$$\max\langle C, X\rangle \text{ such that } A(X) = b,\ X \succeq 0,$$

where $b \in \mathbb{R}^m$ and m is large, for instance $m > 10,000$.

Some ideas:

- Suppose we can split the constraints into two parts so that including only one part makes SDP easy $\rightarrow$ work on partial Lagrangian dual
- Use projection methods
- Spectral Bundle methods

Partial Lagrangian

Now we consider

$$z^* := \max\langle C, X\rangle \text{ such that } A(X) = a,\ B(X) = b,\ X \succeq 0.$$

The idea: Optimizing over $A(X) = a$ without $B(X) = b$ is 'easy', but inclusion of $B(X) = b$ makes SDP difficult. (Could also have inequalities $B(X) \leq b$.)

Partial Lagrangian

Now we consider

$$z^* := \max\langle C, X\rangle \text{ such that } A(X) = a, \; B(X) = b, \; X \succeq 0.$$

The idea: Optimizing over $A(X) = a$ without $B(X) = b$ is 'easy', but inclusion of $B(X) = b$ makes SDP difficult. (Could also have inequalities $B(X) \leq b$.)

Partial Lagrangian Dual (y dual to b):

$$L(X, y) := \langle C, X\rangle + y^T(b - B(X))$$

Dual functional: $(F = \{X : A(X) = a, \; X \succeq 0\})$:

$$f(y) := \max_{X \in F} L(X, y) = b^T y + \max_{x \in F}\langle C - B^T(y), X\rangle$$

Properties of $f(y)$

Recall: $f(y) = b^T y + \max_{x \in F} \langle C - B^T(y), X \rangle$
f is convex (max of linear functions)

Evaluation of $f(y)$ for given y means solving 'simple' SDP.

weak duality: $z^* \leq f(y) \; \forall y$ (holds by construction)

strong duality: $z^* = \min_y f(y)$ (holds under Slater condition)

Basic idea: Minimize $f(y)$ approximately by applying some first order descent methods

Problem: $f(y)$ is nonsmooth (max of linear functions)

Properties of $f(y)$ (2)

Basic assumption: We can compute $f(y)$ easily, yielding also maximizer X^* and $g^* := b - B(X^*)$.

$f(y) = b^T y + \langle C - B^T(y), X^* \rangle = y^T g^* + \langle C, X^* \rangle$, so

$$f(v) \geq v^T g^* + \langle C, X^* \rangle,$$

therefore, using $\langle C, X^* \rangle = f(y) - y^T g^*$ we get

$$f(v) \geq f(y) + \langle g^*, v - y \rangle$$

(This means g^* is subgradient of f at y.)

Thus, evaluating $f(y)$ at y gives function value and subgradient, so use some sort of subgradient optimization to minimize $f(y)$ (at least) approximately.

Minimize f using Bundle Method (2)

Current iterate: $\hat{y}$ with maximizer $\hat{X}$, i.e. $f(\hat{y}) = L(\hat{X}, \hat{y})$. We also assume to have a 'bundle' of other $X_i \in F, \ i = 1, \ldots, k$ with $\hat{X}$ being one of them.
Compute $g_i := b - B(X_i), \phi_i := \langle C, X_i \rangle$.
Using subgradient inequalities for g_i we can minorize f by

$$f(y) \geq l(y) := \max_i \{\langle C, X_i \rangle + \langle g_i, y \rangle\} = \max_{\lambda \in \Lambda} \phi^T \lambda + \langle G\lambda, y \rangle.$$

The key idea:

$$\min_y l(y) + \frac{1}{2t} \|y - \hat{y}\|^2$$

Minimize f using Bundle Method (3)

This is essentially convex quadratic programming in k variables. After exchanging min and max we get:

$$\max_{\lambda \in \Lambda} (\phi + G^T \hat{y})^T \lambda - \frac{t}{2} \|G\lambda\|^2,$$

and new trial point is given by

$$y = \hat{y} - tG\lambda.$$

SDP for Max-Cut + Triangles

As example consider

$$\max\{\langle C, X\rangle : \ diag(X = e), \ X \succeq 0, X \in MET\}$$

$MET = \{X : x_{ij} + x_{ik} + x_{jk} \geq -1, x_{ij} - x_{ik} - x_{jk} \geq -1\}$ asks that X satisfies all the triangle inequalities.
Formally write

$$MET = \{X : B(X) \leq b\}$$

for all $4\binom{n}{3}$ triangle constraints. For $y \geq 0$, we have the partial Lagrangian:

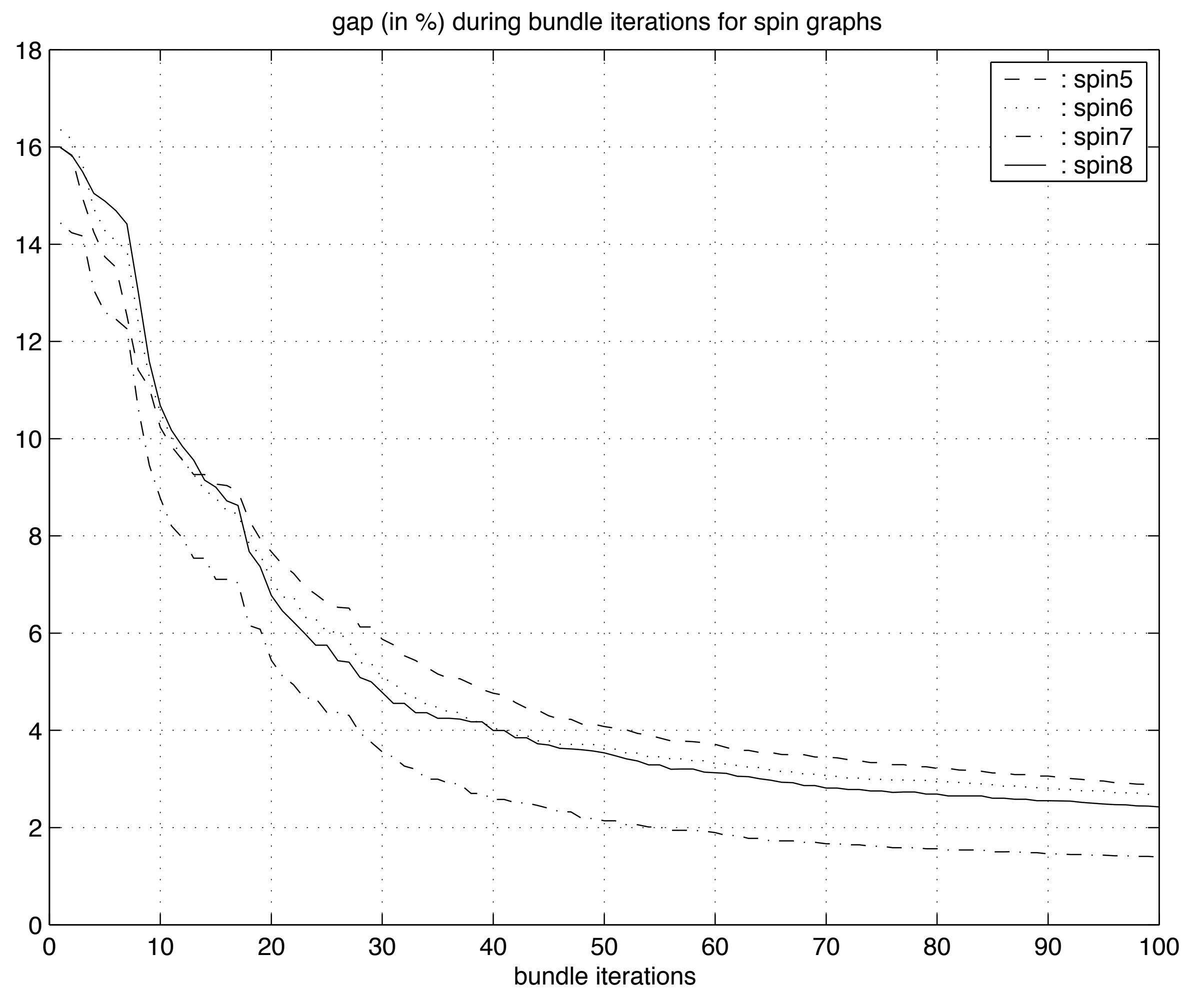
$$f(y) = b^T y + \max\{\langle C - B^T y, X\rangle : X \succeq 0, diag(X) = e\}.$$

A Snapshot

iter	$f(y)$	$\|r\|_1$	$\|r\|_\infty$	contrs. viol.
1	679.3	152541.0	0.96	680822
10	660.4	21132.7	0.73	147094
20	648.1	1234.6	0.52	13605
30	642.2	193.7	0.32	2979
40	639.5	50.8	0.32	957
60	637.6	25.3	0.26	570
80	636.9	17.1	0.23	397
100	636.5	13.5	0.18	369

Max-Cut plus triangles for a graph with $n = 300$. The vector r contains the violation of triangles. Last column has number of violated constraints.

SDP + triangles during Bundle iterations



The gap drops quickly at beginning, then there is tailing off.
Spin Graph instances of order 125 to 512.

Partial Lagrangian: Summary

- first few function evaluations give fast improvent
- tayling off effect (of first order methods)
- high accuracy dificult to achieve