

A (new) input format for conic optimization

Imre Pólik

McMaster University
Advanced Optimization Lab

APMOD2008, Bratislava
May 28, 2008

Outline

- 1 Problem description
 - Conic optimization
 - Semidefinite optimization
 - Special problems
- 2 Requirements
 - Existing input formats
 - General requirements
 - Modelling requirements
 - Possible routes
- 3 The proposed format
 - Layout
 - Declarations
 - Data

General conic optimization

$$\min c^T x$$

$$Ax = b$$

$$x \in \mathcal{K}$$

$$\max b^T y$$

$$A^T y + s = c$$

$$s \in \mathcal{K}^*$$

The cone \mathcal{K} can be

Linear: $x \geq 0$

Second-order: $x_0 \geq \|x\|_2$

Rotated second-order: $x_0 x_1 \geq \|x_{2:n}\|$, and $x_0 \geq 0$

Semidefinite: x is (can be assembled into) a symmetric,
positive semidefinite matrix, or a

product of these.

Semidefinite optimization

- Standard form

$$\begin{array}{ll} \min C \bullet X & \max b^T y \\ \mathcal{A}X = b & \mathcal{A}^*y + S = C \\ X \succeq 0 & S \succeq 0, \end{array} \quad (\text{P-D})$$

where $b, y \in \mathbb{R}^m, X, S, C \in \mathbb{R}^{n^2}, \mathcal{A} : \mathbb{R}^{n^2} \rightarrow \mathbb{R}^m$

Semidefinite optimization

- Standard form

$$\begin{array}{ll}
 \min C \bullet X & \max b^T y \\
 \mathcal{A}X = b & \mathcal{A}^*y + S = C \\
 X \succeq 0 & S \succeq 0,
 \end{array} \quad (\text{P-D})$$

where $b, y \in \mathbb{R}^m$, $X, S, C \in \mathbb{R}^{n^2}$, $\mathcal{A} : \mathbb{R}^{n^2} \rightarrow \mathbb{R}^m$

- Linear operator \mathcal{A}

$$\begin{aligned}
 \mathcal{A}X &= (A_i \bullet X)_{i=1}^m \\
 \mathcal{A}^*y &= \sum_{i=1}^m A_i y_i
 \end{aligned}$$

\Rightarrow too restrictive

Special forms

- Rank one, low rank A_i

$$A_i = aa^T, A_i \bullet X = a^T X a$$

- can be exploited inside the IPM
- cannot be recovered exactly from A_i

Special forms

- Rank one, low rank A_i

$$A_i = aa^T, \quad A_i \bullet X = a^T X a$$

- can be exploited inside the IPM
- cannot be recovered exactly from A_i

- General operators

$$\mathcal{A}X = AX + XA, \text{ or}$$

$$\mathcal{A}X = AXB + BXA$$

- \mathcal{A} is a large Kronecker product
- huge savings in storage and computation
- one needs to have \mathcal{A}^*
- right-hand side is a matrix!

Special forms

- Rank one, low rank A_i

$$A_i = aa^T, A_i \bullet X = a^T X a$$

- can be exploited inside the IPM
- cannot be recovered exactly from A_i

- General operators

$$\mathcal{A}X = AX + XA, \text{ or}$$

$$\mathcal{A}X = AXB + BXA$$

- \mathcal{A} is a large Kronecker product
- huge savings in storage and computation
- one needs to have \mathcal{A}^*
- right-hand side is a matrix!
- Cone intersections
 - $X \succeq 0, X \geq 0$

Special forms

- Rank one, low rank A_i

$$A_i = aa^T, A_i \bullet X = a^T X a$$

- can be exploited inside the IPM
- cannot be recovered exactly from A_i

- General operators

$$\mathcal{A}X = AX + XA, \text{ or}$$

$$\mathcal{A}X = AXB + BXA$$

- \mathcal{A} is a large Kronecker product
 - huge savings in storage and computation
 - one needs to have \mathcal{A}^*
 - right-hand side is a matrix!
- Cone intersections
 - $X_1 \succeq 0, X_2 \geq 0, X_1 = X_2$

Special forms

- Rank one, low rank A_i

$$A_i = aa^T, \quad A_i \bullet X = a^T X a$$

- can be exploited inside the IPM
- cannot be recovered exactly from A_i

- General operators

$$\mathcal{A}X = AX + XA, \text{ or}$$

$$\mathcal{A}X = AXB + BXA$$

- \mathcal{A} is a large Kronecker product
- huge savings in storage and computation
- one needs to have \mathcal{A}^*
- right-hand side is a matrix!
- Cone intersections
 - $X_1 \succeq 0, X_2 \geq 0, X_1 = X_2$
 - $0 \preceq X \preceq U$

Special forms

- Rank one, low rank A_i

$$A_i = aa^T, \quad A_i \bullet X = a^T X a$$

- can be exploited inside the IPM
- cannot be recovered exactly from A_i

- General operators

$$\mathcal{A}X = AX + XA, \text{ or}$$

$$\mathcal{A}X = AXB + BXA$$

- \mathcal{A} is a large Kronecker product
- huge savings in storage and computation
- one needs to have \mathcal{A}^*
- right-hand side is a matrix!
- Cone intersections
 - $X_1 \succeq 0, X_2 \geq 0, X_1 = X_2$
 - $0 \preceq X_1 = X_2 \preceq U$
 - n^2 new equalities and variables!

Existing input formats

- Mixed problems
 - SeDuMi, SDPT3, SDPpack
 - Matlab-based, binary
- SDP
 - Sparse SDPA format
 - Extended SDPA formats
 - SDPLR (low rank data matrices)
 - PCSDP (rank-1 data matrices)
 - Graph problems (DSDP)
- SOCP
 - MOSEK
 - LOQO
 - CPLEX
- Modelling languages
 - CVX
 - Yalmip

General requirements

- Portability
 - text-based
- *Reasonable* compatibility with existing formats
- Easily generated/modified by the user
- Powerful modelling capabilities
- Straightforward processing
- In-place preprocessing

Modelling requirements

- Mixed linear/second-order/semidefinite optimization
- Cones in arbitrary order
- Sparse and dense representations
- Rank-one/low rank constraint matrices
- General linear operators
- Cone intersections
- Indirectly defined problems
 - graph problems
 - robust problems
- More cones
 - power cones
 - exponential cone
- More objectives
 - quadratic
 - polynomial
 - log det

Potential candidates

- Modelling languages
 - CVX
 - YALMIP
- XML
 - COIN-OS
 - only feasible together with a modelling environment
- A new format

The proposed format – problem layout

| | | |
|-----------|------------------|----------------------------------|
| F 2 | L- 4 | S 3 |
| $x_{1:2}$ | $x_{3:7} \leq 0$ | $\text{mat}(x_{8:16}) \succeq 0$ |

| | | | | | | | |
|--------------------|--------------------|--------------------|---|---|-----------|---|------|
| \mathcal{C}_1 | \mathcal{C}_2 | \mathcal{C}_3 | = | 3 | 0- | | |
| \mathcal{A}_{11} | \mathcal{A}_{12} | \mathcal{A}_{13} | | | \preceq | 1 | E 1 |
| \mathcal{A}_{21} | \mathcal{A}_{22} | \mathcal{A}_{23} | | | \succeq | 0 | S- 3 |
| \mathcal{A}_{31} | \mathcal{A}_{32} | \mathcal{A}_{33} | | | | | L 2 |

- Declare variables and constraints
- Define the $\mathcal{C}_j, \mathcal{A}_{ij}$ mappings and the RHS
 - sparse: only the nonzero blocks
 - several choices for one block (sparse, dense, rank-one, operator, etc.)

The proposed format – declarations

- Variables in arbitrary order
 - F, L, Q, R, S and more (SL, RL, B, I, E)

F 3 L 5 S 7 7 7 L 2 R 5 F 2 S 5

The proposed format – declarations

- Variables in arbitrary order
 - F, L, Q, R, S and more (SL, RL, B, I, E)
 - repetition

F 3 L 5 S 7x3 L 2 R 5 F 2 S 5

The proposed format – declarations

- Variables in arbitrary order
 - F, L, Q, R, S and more (SL, RL, B, I, E)
 - repetition
 - negative cones: S-, L-

F 3 L 5 S 7x3 L 2 R 5 F 2 S 5

The proposed format – declarations

- Variables in arbitrary order
 - F, L, Q, R, S and more (SL, RL, B, I, E)
 - repetition
 - negative cones: S-, L-

F 3 L 5 S 7x3 L 2 R 5 F 2 S 5

- Constraints in arbitrary order
 - E, L, Q, R, S and more (SL, RL, B, I, O)
 - dual variables
 - objective is declared with O+, O-
 - repetition
 - “less than” S-, L-

E 5 L 5 S 3 S- 5

The proposed format – problem data

- Vectors
 - Dense vector
D 1 8 4 -6 8 3 9.2
 - Sparse vector
S 3, 1.3 2.8 -4, 1 13 20
- Matrices are treated similarly
- Special \mathcal{A}_{ij} blocks: dictionary

| Code | Form | Data |
|--------|---------------------------|-------------|
| D or S | generic block | the matrix |
| LD | $\log \det(X)$ | \emptyset |
| A1 | $AX + XA$ | A |
| A2 | $AX + XA^T$ | A |
| A3 | $AXB + BXA$ | A, B |
| L1 | $\mathcal{A}_{ij} = aa^T$ | a |

...

2 3 L1

D 1 8 4 -6 8 3 9.2

Conclusions

- Barebone modelling language
- Little overhead on the data
- Similar to existing formats
- Flexible
- Easily converted to XML
- Extends to more general cones
- Work in progress (pre alpha)
- To be featured in pSeDuMi

A (new) input format for conic optimization

Imre Pólik

McMaster University
Advanced Optimization Lab

APMOD2008, Bratislava
May 28, 2008