SeDuMi: a package for conic optimization

Imre Pólik, Tamás Terlaky, Yuriy Zinchenko
McMaster University, Advanced Optimization Lab

Introduction

What is SeDuMi?

- Optimization over symmetric cones
  - linear, second order, semidefinite
  - complex variables
- Interior point method
  - primal-dual predictor-corrector scheme
  - self dual embedding
- Open source: GPL, written in Matlab and C
- Modelling languages:
  - YALMIP (Johan Löfberg)
  - CVX (Michael C. Grant)
  - Gloptipoly (Didier Henrion)
  - SOSTools (Stephen Prajna et al.)
- Widely used: both industry and academics
- Advantages: very high numerical accuracy, robustness, efficient sparse system handling, mixed second-order/semidefinite problems, Matlab
- Weaknesses: large dense problems, memory requirements, Matlab

History

late 1997: Jos F. Sturm starts SeDuMi
summer 1998: SeDuMi 1.0
November 2002: SeDuMi 1.05R5 (last version by Jos)
November 2003: Jos dies
October 2004: AdvOL at McMaster takes over
June 2005: SeDuMi 1.1 (new version)
October 2006: SeDuMi 1.1R2 (latest version)

Theoretical background

- Primal-dual conic optimization

$$\begin{array}{lll} \min c^T x & \max b^T y & \\ Ax = b & A^T y + s = c & \text{(CO)} \\ x \in \mathcal{K} & s \in \mathcal{K} & \end{array}$$

where $x, c, s \in \mathbb{R}^n$, $A \in \mathbb{R}^{m \times n}$, $b, y \in \mathbb{R}^m$ and $\mathcal{K} \subset \mathbb{R}^m$ is a closed, convex, pointed, solid, self-dual cone

- Primal-dual path-following method
- Predictor-corrector scheme
- Self-dual embedding
- Complexity: $\mathcal{O}\left(\sqrt{n}\log\frac{n}{\varepsilon}\right)$ to find an ε-optimal solution

Modelling languages

GAMS, AMPL, AIMMS, LINDO, etc.

- None of the major modelling languages supports SDP
- User input required

YALMIP

- Very advanced and developed
  - More general problems (SDP, NLP, MILP), 30+ solvers
  - Automatic dualization, robustification, convexity analysis
- Widely used for optimal control problems

Lyapunov stability:

$$P \succeq 0$$
$$A^T P + PA \preceq 0$$

```
A = [-1 2 0;-3 -4 1;0 0 -2];
P = sdpvar(3,3);
F = set(P>=0)+set(A'*P+P*A<=0);
F = F + set(trace(P) == 1);
solvesdp(F);
```

CVX

- Disciplined Convex Programming
- Least squares, LP, QP, GP, NLP is planned
- Currently uses only SeDuMi, more solvers will be added

Norm optimization:

$$\begin{array}{l} \min \|Ax-b\|_2 \\ Cx = d \\ \|x\|_\infty \le 0.4 \end{array}$$

```
m = 20; n = 10; p = 4;
A = randn(m,n); b = randn(m,1);
C = randn(p,n); d = randn(p,1);
cvx_begin
 variable x(n)
 minimize( norm( A*x-b, 2 ) )
 subject to
 C * x == d;
 norm( x, Inf ) <= 0.4;
cvx_end
```

Gloptipoly

- Polynomial systems, nonconvex problems
- Relaxations, bounds, global optimality detection

$$\begin{array}{c} \max x_1^2 + x_2^2 \\ 2x_1^2 + 3x_2^2 + 2x_1x_2 \le 1 \\ 3x_1^2 + 2x_2^2 - 4x_1x_2 \le 1 \end{array}$$

```
P1 = [0 0 -1; 0 0 0; -1 0 0];
P2 = [1 0 -3; 0 -2 0; -2 0 0];
P3 = [1 0 -2; 0 4 0; -3 0 0];
gloptipoly(P);
```

SOSTOOLS

- Sum-of-squares optimization
- Univariate polynomial minimization
- SDP relaxation for semialgebraic problems

SOS representation for:
$p(x) = 4x^4y^6 + x^2 - xy^2 + y^2$

```
syms x y;
p = 4*x^4*y^6+x^2-x*y^2+y^2;
[Q,Z]=findsos(p,'rational');
```

Examples are taken from the official websites.

Current projects

Parallelization/Improved linear algebra

- BLAS/LAPACK/ScaLAPACK
  - ACML, Intel MKL, ATLAS
  - matrix products/factorizations
  - automatic parallelization
  - linear speedup
- OpenMP
  - compilers: Pathscale, IBM, Intel, GCC (from 4.2)
  - forming the normal equation
  - very good speedup
- Better linear algebra
  - more efficient sparse/dense handling

Preprocessing

- Even elementary techniques work (identical/opposite sign/fixed variables, redundant constraints)
- Finding block-diagonal structure
- Decomposing narrow-band matrices
- Conic decomposition (Kojima et. al, Plaza Martínez/Krishnan, Young/Anjos)

Adaptive techniques

- Optimal parameter selection
  - tuning on a small problem, testing on the large one
  - significant savings within a problem group
- Change parameters during the iterations (corrector type, update method, neighbourhood parameters, step differentiation)
  - Online learning methods, reinforcement learning
- Starting point selection

Advanced infeasibility detection

- Early identification of infeasibility to save time
  - modelling errors
  - unrealistic designs
- Change the algorithm if infeasibility is suspected
- Full infeasibility analysis
  - Detect contradicting constraints
  - Handle weak infeasibility

Future plans

Reimplementation

- Limitations of Matlab
  - Price
  - Difficult to embed in another application
  - Memory limitations
- Our goal is to make/keep SeDuMi
  - free
  - platform-independent
  - efficient
  - modular
  - embeddable
- Candidate languages:
  - C/C++
  - Python (with SciPy and NumPy)
  - Octave

Bibliography

[1] A. Ben-Tal and A. Nemirovski. *Lectures on Modern Convex Optimization: Analysis, Algorithms, and Engineering Applications*. MPS-SIAM Series on Optimization. SIAM, Philadelphia, PA, 2001.

[2] M. Grant, S. Boyd, and Y. Ye. Disciplined convex programming. In Leo Liberti and Nelson Maculan, editors, *Global Optimization: From Theory to Implementation*, Nonconvex Optimization and its Applications. Kluwer, Dordrecht, 2005.

[3] M. Grant, S. Boyd, and Y. Ye. *CVX Users' Guide*, 2007.

[4] D. Henrion and J. B. Lasserre. GloptiPoly: Global optimization over polynomials with Matlab and SeDuMi. *ACM Transactions on Mathematical Software*, 29(2):165–194, June 2003.

[5] J. Löfberg. YALMIP: A toolbox for modeling and optimization in MATLAB. In *Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004.

[6] I. Pólik and T. Terlaky. Detecting infeasibility in conic optimization: Theory and practice. Technical report, McMaster University, Advanced Optimization Lab, 2007. In preparation.

[7] S. Prajna, A. Papachristodoulou, P. Seiler, and P. A. Parrilo. *SOSTOOLS: Sum of squares optimization toolbox for MATLAB*, 2004.

[8] J. F. Sturm. *Primal-Dual Interior Point Approach to Semidefinite Programming*. Phd thesis, Tinbergen Institute Research Series vol. 156, Tilburg University, 1997.

[9] J.F. Sturm. Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11–12:625–653, 1999. Special issue on Interior Point Methods (CD supplement with software).

Acknowledgements: The authors were supported by the NSERC Discovery Grant #5-48923, the Canada Research Chair Program and a MITACS project.