SCIP-SDP: A Framework for Solving Mixed-Integer Semidefinite Programs

Tristan Gally

joint work with Marc E. Pfetsch and Stefan Ulbrich

SFB 805

Control of Uncertainty in Load-Carrying Structures in Mechanical Engineering

Mixed-Integer Semidefinite Programming

TECHNISCHE UNIVERSITÄT DARMSTADT

- Mixed-Integer Semidefinite Program

MISDP

$$\begin{aligned} \inf \quad & b^T y \\ \text{s.t.} \quad & \sum_{i=1}^{m} A_i y_i - C \succeq 0, \\ & y_i \in \mathbb{Z} \quad \forall\, i \in \mathcal{I} \end{aligned}$$

for symmetric matrices A_i, C.

Mixed-Integer Semidefinite Programming

TECHNISCHE UNIVERSITÄT DARMSTADT

- Mixed-Integer Semidefinite Program

MISDP

$$\begin{aligned} \inf \quad & b^T y \\ \text{s.t.} \quad & \sum_{i=1}^{m} A_i y_i - C \succeq 0, \\ & y_i \in \mathbb{Z} \quad \forall\, i \in \mathcal{I} \end{aligned}$$

for symmetric matrices A_i, C.

- Linear constraints, variable bounds and multiple blocks possible within the SDP-constraint.

Mixed-Integer Semidefinite Programming

- Mixed-Integer Semidefinite Program

MISDP

$$\begin{aligned} \inf \quad & b^T y \\ \text{s.t.} \quad & \sum_{i=1}^{m} A_i y_i - C \succeq 0, \\ & \qquad\qquad y_i \in \mathbb{Z} \quad \forall\, i \in \mathcal{I} \end{aligned}$$

for symmetric matrices A_i, C.

- Linear constraints, variable bounds and multiple blocks possible within the SDP-constraint.
- Approach: Nonlinear branch-and-bound using interior-point SDP-solvers.

Applications

- Robust Truss Topology Design
- Cardinality Constrained Least Squares
- Minimum k-Partitioning
- Compressed Sensing
- Optimal Transmission Switching Problem in Electrical Grids
- ...

Contents

Strong Duality in SDP

TECHNISCHE
UNIVERSITÄT
DARMSTADT

Dual SDP (D)

$$\begin{aligned} \inf \quad & b^T y \\ \text{s.t.} \quad & \sum_{i=1}^{m} A_i y_i - C \succeq 0 \\ & y \in \mathbb{R}^m \end{aligned}$$

Primal SDP (P)

$$\begin{aligned} \sup \quad & C \bullet X \\ \text{s.t.} \quad & A_i \bullet X = b_i \quad \forall\, i \le m \\ & X \succeq 0 \end{aligned}$$

Strong Duality in SDP

TECHNISCHE UNIVERSITÄT DARMSTADT

Dual SDP (D)

$$\begin{aligned} \inf \quad & b^T y \\ \text{s.t.} \quad & \sum_{i=1}^{m} A_i y_i - C \succeq 0 \\ & y \in \mathbb{R}^m \end{aligned}$$

Primal SDP (P)

$$\begin{aligned} \sup \quad & C \bullet X \\ \text{s.t.} \quad & A_i \bullet X = b_i \quad \forall\, i \leq m \\ & X \succeq 0 \end{aligned}$$

- Strong duality does not hold in general for (P) and (D).

Strong Duality in SDP

TECHNISCHE UNIVERSITÄT DARMSTADT

Dual SDP (D)

$$\begin{aligned} \inf \quad & b^T y \\ \text{s.t.} \quad & \sum_{i=1}^{m} A_i y_i - C \succeq 0 \\ & y \in \mathbb{R}^m \end{aligned}$$

Primal SDP (P)

$$\begin{aligned} \sup \quad & C \bullet X \\ \text{s.t.} \quad & A_i \bullet X = b_i \quad \forall\, i \leq m \\ & X \succeq 0 \end{aligned}$$

- Strong duality does not hold in general for (P) and (D).
- If Slater condition holds for (P) or (D), i.e., there exists a feasible $X \succ 0$ for (P) or y such that $\sum_{i=1}^{m} A_i y_i - C \succ 0$ in (D), then strong duality holds.

Strong Duality in SDP

Dual SDP (D)

$$\begin{aligned} \inf \quad & b^T y \\ \text{s.t.} \quad & \sum_{i=1}^{m} A_i y_i - C \succeq 0 \\ & y \in \mathbb{R}^m \end{aligned}$$

Primal SDP (P)

$$\begin{aligned} \sup \quad & C \bullet X \\ \text{s.t.} \quad & A_i \bullet X = b_i \quad \forall\, i \le m \\ & X \succeq 0 \end{aligned}$$

- Strong duality does not hold in general for (P) and (D).
- If Slater condition holds for (P) or (D), i.e., there exists a feasible $X \succ 0$ for (P) or y such that $\sum_{i=1}^{m} A_i y_i - C \succ 0$ in (D), then strong duality holds.
- If Slater holds for (P), optimal objective of (D) is attained and vice versa.

Strong Duality in SDP

TECHNISCHE UNIVERSITÄT DARMSTADT

Dual SDP (D)

$$\begin{aligned} \inf \quad & b^T y \\ \text{s.t.} \quad & \sum_{i=1}^{m} A_i y_i - C \succeq 0 \\ & y \in \mathbb{R}^m \end{aligned}$$

Primal SDP (P)

$$\begin{aligned} \sup \quad & C \bullet X \\ \text{s.t.} \quad & A_i \bullet X = b_i \quad \forall\, i \leq m \\ & X \succeq 0 \end{aligned}$$

- Strong duality does not hold in general for (P) and (D).
- If Slater condition holds for (P) or (D), i.e., there exists a feasible $X \succ 0$ for (P) or y such that $\sum_{i=1}^{m} A_i y_i - C \succ 0$ in (D), then strong duality holds.
- If Slater holds for (P), optimal objective of (D) is attained and vice versa.
- Existence of a KKT-point is only guaranteed if Slater holds for both. This is assumed by most SDP-solvers.

Discrete Optimization

Strong Duality in Branch-and-Bound

TECHNISCHE UNIVERSITÄT DARMSTADT

Theorem

Let (D_+) be the problem formed by adding a linear constraint to (D). If

- strong duality holds for (P) and (D)
- the set of optimal $Z := \sum_{i=1}^{m} A_i y_i - C$ in (D) is compact
- the problem (D_+) is feasible

then strong duality also holds for (D_+) and (P_+) and the set of optimal Z for (D_+) is compact.

Strong Duality in Branch-and-Bound

TECHNISCHE UNIVERSITÄT DARMSTADT

Theorem

Let (D_+) be the problem formed by adding a linear constraint to (D). If

- strong duality holds for (P) and (D)
- the set of optimal $Z := \sum_{i=1}^{m} A_i y_i - C$ in (D) is compact
- the problem (D_+) is feasible

then strong duality also holds for (D_+) and (P_+) and the set of optimal Z for (D_+) is compact.

Proposition

If (P) satisfies the Slater condition and $A_1, \ldots, A_m$ are linearly independent, then the Slater condition also holds for (P_+).

Strong Duality in Branch-and-Bound

TECHNISCHE UNIVERSITÄT DARMSTADT

Theorem

Let (D_+) be the problem formed by adding a linear constraint to (D). If

- strong duality holds for (P) and (D)
- the set of optimal $Z := \sum_{i=1}^{m} A_i y_i - C$ in (D) is compact
- the problem (D_+) is feasible

then strong duality also holds for (D_+) and (P_+) and the set of optimal Z for (D_+) is compact.

Proposition

If (P) satisfies the Slater condition and $A_1, \ldots, A_m$ are linearly independent, then the Slater condition also holds for (P_+).

- Slater condition in (D_+) and existence of a KKT-point may get lost.

Contents

Solving the SDP-Relaxations

TECHNISCHE UNIVERSITÄT DARMSTADT

- SDP-relaxations in B&B-tree are solved via interior-point methods.
- If interior-point solver did not converge for original formulation, solve

Feasibility Check

$$\begin{aligned} \inf \quad & r \\ \text{s.t.} \quad & \sum_{i=1}^{m} A_i\, y_i - C + I\, r \succeq 0. \end{aligned}$$

If $r^* > 0$, original problem is infeasible and node can be cut off.

Solving the SDP-Relaxations

TECHNISCHE UNIVERSITÄT DARMSTADT

- If problem is not infeasible, solve

Penalty Formulation

$$\begin{aligned} \inf \quad & b^\top y + \Gamma\, r \\ \text{s.t.} \quad & \sum_{i=1}^{m} A_i\, y_i - C + I\, r \succeq 0, \\ & r \geq 0 \end{aligned}$$

for sufficiently large Γ to compute a lower bound.

- If $r^* = 0$, then solution is also optimal for original problem.

Dual Fixing

- Generalization of reduced-cost fixing for MILPs.
- Used for interior-point LP-solvers by Mitchell (1997), primal MISDPs by Helmberg (2000) and general MINLPs by Vigerske (2012).

Dual Fixing

TECHNISCHE UNIVERSITÄT DARMSTADT

- Generalization of reduced-cost fixing for MILPs.
- Used for interior-point LP-solvers by Mitchell (1997), primal MISDPs by Helmberg (2000) and general MINLPs by Vigerske (2012).

Theorem

- (X, W, V): primal feasible solution, where W, V are primal variables corresponding to variable bounds ℓ, u in the dual
- f: corresponding primal objective value
- U: upper bound on the optimal objective value of the MISDP

Then for every optimal solution of the MISDP

$$y_j \leq \ell_j + \frac{U - f}{W_{jj}} \quad \text{if } \ell_j > -\infty \qquad \text{and} \qquad y_j \geq u_j - \frac{U - f}{V_{jj}} \quad \text{if } u_j < \infty.$$

- If $U - f < W_{jj}$ for binary y_j, it can be fixed to 0, if $U - f < V_{jj}$, then $y_j = 1$.

Heuristics & Branching

TECHNISCHE UNIVERSITÄT DARMSTADT

Heuristics

- Diving: Iteratively round variables and resolve SDP-relaxation.
- Randomized rounding: Round all binary variables with probability to round up equal to relaxation value. Resolve SDP for remaining continuous variables.

Heuristics & Branching

Heuristics

- Diving: Iteratively round variables and resolve SDP-relaxation.
- Randomized rounding: Round all binary variables with probability to round up equal to relaxation value. Resolve SDP for remaining continuous variables.

Branching

- Most fractional
- Highest absolute objective coefficient
- Product of fractionality and objective
- Inference: Number of implied fixings for this branching in the past.

Contents

Test Environment

TECHNISCHE UNIVERSITÄT DARMSTADT

- Testset
  - 60 truss topology instances
  - 65 cardinality constrained least squares instances
    - including 20 real-world instances from cancer detection
  - 69 minimum k-partitioning instances
    - including 10 real-world instances for VLSI chip design

Test Environment

TECHNISCHE UNIVERSITÄT DARMSTADT

- Testset
  - 60 truss topology instances
  - 65 cardinality constrained least squares instances
    - including 20 real-world instances from cancer detection
  - 69 minimum *k*-partitioning instances
    - including 10 real-world instances for VLSI chip design
- Preliminary version of SCIP-SDP 2.1.0 and developer version of SCIP 3.2.1
- Free for academic use
- DSDP 5.8 or SDPA 7.3.8
- Time limit of 3600 seconds
- Shifted geometric means, times in seconds
- Linux cluster with Intel i3 CPUs with 3.2GHz, 4MB cache and 8GB memory

Slater Condition in Practice

TECHNISCHE UNIVERSITÄT DARMSTADT

Portion of SDPs satisfying Slater condition

	Dual Slater				Primal Slater		
problem	✓	✗	inf	?	✓	✗	?
TTD	91.04 %	4.57 %	4.38 %	0.01 %	98.91 %	0.00 %	1.09 %
CLS	80.32 %	1.24 %	17.83 %	0.61 %	100.00 %	0.00 %	0.00 %
M*k*-P	2.56 %	94.39 %	1.09 %	1.96 %	100.00 %	0.00 %	0.00 %
Overall	55.98 %	35.40 %	7.71 %	0.91 %	99.66 %	0.00 %	0.34 %

Discrete Optimization

SDP-Solvers depending on Slater Condition

TECHNISCHE UNIVERSITÄT DARMSTADT

Behaviour if Slater condition holds for (P) and (D)

solver	number	default	penalty	bound	unsucc
DSDP	976,547	99.16 %	0.39 %	0.00 %	0.46 %
SDPA	749,151	99.96 %	0.01 %	0.00 %	0.03 %

Behaviour if Slater condition fails for (P) or (D)

solver	number	default	penalty	bound	unsucc
DSDP	55,791	67.72 %	0.11 %	0.01 %	32.15 %
SDPA	32,616	60.19 %	1.66 %	29.05 %	9.10 %

Behaviour if problem is infeasible

solver	number	default	penalty	bound	unsucc
DSDP	51,508	58.14 %	41.85 %	0.00 %	0.00 %
SDPA	53,088	30.75 %	69.25 %	0.00 %	0.00 %

Influence of SDP-Solver and Branching Rule

TECHNISCHE UNIVERSITÄT DARMSTADT

Solving times for different SDP-solvers and branching rules

settings	solved	nodes	time
`DSDP-infer`	120	522.5	786.9
`DSDP-infobj`	138	173.4	556.5
`DSDP-obj`	129	233.2	699.8
`DSDP-inf`	127	234.1	749.1
`SDPA-infer`	122	488.8	584.8
`SDPA-infobj`	136	202.5	394.4
`SDPA-obj`	136	248.1	455.6
`SDPA-inf`	120	274.3	540.8

Influence of Dual Fixing and Heuristics

TECHNISCHE UNIVERSITÄT DARMSTADT

Solving times for different settings using SDPA and infobj branching

settings	solved	nodes	time
`noheur`	141	261.6	350.3
`noheur-dualfix`	148	261.0	319.9
`dive_rootnode`	136	243.7	394.4
`dive_rootnode-dualfix`	144	232.9	339.8
`dive_depth10`	125	240.7	537.6
`dive_depth10-dualfix`	139	197.6	395.3
`rand_rootnode`	142	254.8	339.5
`rand_rootnode-dualfix`	149	247.4	261.9
`rand_depth10`	146	252.1	316.2
`rand_depth10-dualfix`	156	218.3	228.5

SCIP-SDP is available in source code at

`http://www.opt.tu-darmstadt.de/scipsdp/`

Thank you for your attention!

References

TECHNISCHE UNIVERSITÄT DARMSTADT

Tobias Achterberg.
Constraint Integer Programming.
PhD thesis, TU Berlin, 2007.

Aharon Ben-Tal and Arkadii Nemirovski.
Robust truss topology design via semidefinite programming.
SIAM Journal on Optimization, 7(4):991–1016, 1997.

Steven J. Benson and Yinyu Ye.
Algorithm 875: DSDP5–software for semidefinite programming.
ACM Transactions on Mathematical Software, 34(4):Article 16, 20 pages, 2008.

Andreas Eisenblätter.
The semidefinite relaxation of the k-partition polytope is strong.
In W. J. Cook and A. S. Schulz, editors, *Proceedings of the 9th International IPCO Conference on Integer Programming and Combinatorial Optimization*, volume 2337 of *Lecture Notes in Computer Science*, pages 273–290. Springer, Berlin Heidelberg, 2002.

Tristan Gally, Marc E. Pfetsch, and Stefan Ulbrich.
A framework for solving mixed-integer semidefinite programs.
Technical report, Optimization Online, 2016.

Christoph Helmberg.
Fixing variables in semidefinite relaxations.
SIAM Journal on Matrix Analysis and Applications, 21(3):952–969, 2000.

Sonja Mars.
Mixed-Integer Semidefinite Programming with an Application to Truss Topology Design.
PhD thesis, FAU Erlangen-Nürnberg, 2013.

John E. Mitchell.
Fixing variables and generating classical cutting planes when using an interior point branch and cut method to solve integer programming problems.
European Journal of Operational Research, 97:139–148, 1997.

Mert Pilanci, Martin J. Wainwright, and Laurent El Ghaoui.
Sparse learning via Boolean relaxations.
Mathematical Programming Series B, 151(1):62–87, 2015.

Stefan Vigerske.
Decomposition in Multistage Stochastic Programming and a Constraint Integer Programming Approach to Mixed-Integer Nonlinear Programming.
PhD thesis, HU Berlin, 2012.

Makoto Yamashita, Katsuki Fujisawa, Kazuhide Nakata, Maho Nakata, Mituhiro Fukuda, Kazuhiro Kobayashi, and Kazushige Goto.
A high-performance software package for semidefinite programs: SDPA 7.
Research Report B-460, Dept. of Mathematical and Computing Science, Tokyo Institute of Technology, September 2010.