Developing spatial branch & bound solvers

For mixed-integer nonlinear optimization

Radu Baltean-Lugojan

Francesco Ceccon

Jan Kronqvist

Ruth Misener

Miten Mistry

Computational Optimisation Group
Department of Computing

Twitter https://twitter.com/CogImperial

Imperial College London

computational optimisation group

EPSRC
Pioneering research and skills

EPSRC EP/P016871/1

2021/06/23

Outline

Outline

Definition: Mixed-integer nonlinear optimization (MINLP)[1]

Grossmann and Sargent [1979]

This class of problem is very difficult to solve, and no general method yet exists for its efficient solution.

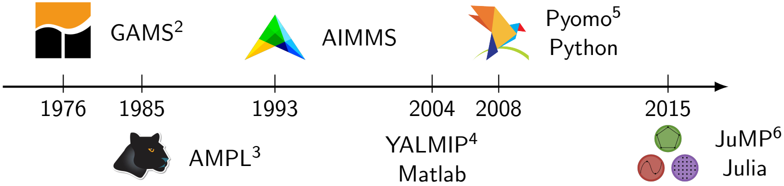
$$
\begin{aligned}
\min_{\mathbf{x}\in X,\, \mathbf{y}\in Y} \quad & f(\mathbf{x}, \mathbf{y}) && \longleftarrow \text{Objective function} \\
\text{s.t.} \quad & \mathbf{h}(\mathbf{x}, \mathbf{y}) = \mathbf{0} && \longleftarrow \text{Equality constraints} \\
& \mathbf{g}(\mathbf{x}, \mathbf{y}) \leq \mathbf{0} && \longleftarrow \text{Inequality constraints} \\
& \mathbf{x} \in \left[\mathbf{x}^{\mathrm{L}}, \mathbf{x}^{\mathrm{U}}\right] \subset \mathbb{R}^{n_x} && \longleftarrow \text{Continuous variable bounds} \\
& y \in \{0, 1\}^{n_y} && \longleftarrow \text{Binary variables}
\end{aligned}
$$

Assumptions

$f, \mathbf{h}, \mathbf{g} \in \mathcal{C}^2$

[1] Grossmann and Biegler [2004], Floudas and Gounaris [2009], Bussieck and Vigerske [2010], D'Ambrosio and Lodi [2011], Belotti et al. [2013], Boukouvala et al. [2016]

How can we represent MINLP?



Many MINLP solvers also have dedicated interfaces

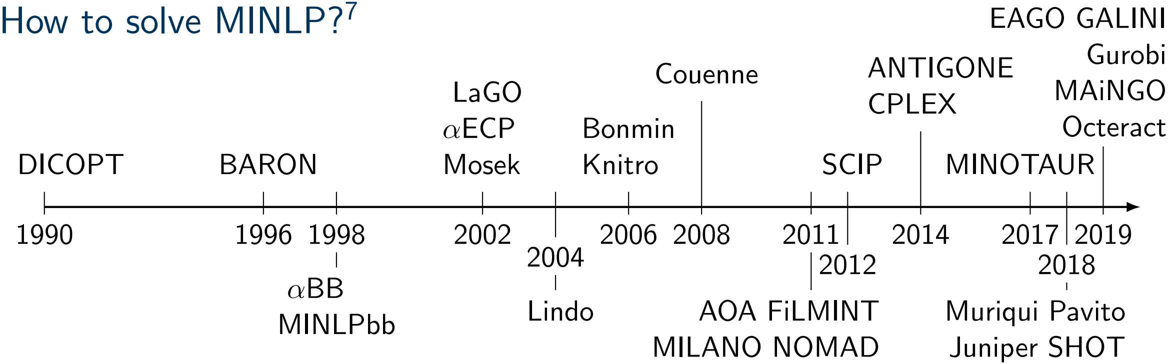
Mosel for FICO Xpress, ZIMLP for SCIP [Koch, 2004], etc.

General purpose mathematical software with MINLP connections

Excel, Matlab, R, . . .

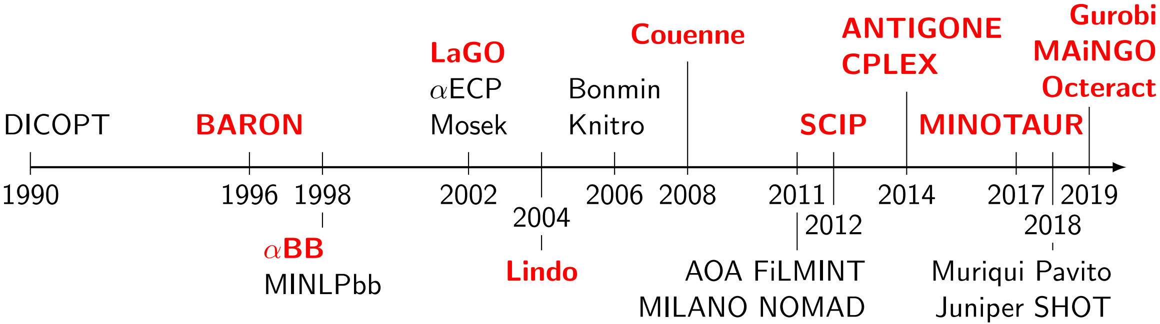
[2]Brooke et al. [1997], [3]Fourer et al. [1993], [4]Löfberg [2004], [5]Hart et al. [2017], [6]Dunning et al. [2017]

How to solve MINLP?[7]



[7] α**BB** Adjiman et al. [1998a] • α**ECP** Westerlund and Pörn [2002] • **ANTIGONE** Misener and Floudas [2014] • **AOA** Hunting [2011] • **Baron** Tawarmalani and Sahinidis [2005] • **Bonmin** Bonami et al. [2008] • **Couenne** Belotti et al. [2009] • **CPLEX** Bonami et al. [2016] • **DICOPT** Viswanathan and Grossmann [1990] • **EAGO** Wilhelm and Stuber [2017] • **FilMINT** Abhishek et al. [2010] • **GALINI** • **Gurobi** • **Juniper** Kröger et al. [2018] • **Knitro** Byrd et al. [2006] • **LaGO** Nowak et al. [2003] • **LINDO** Gau and Schrage [2004] • **MAiNGO** Najman and Mitsos [2019] • **MILANO** Benson [2011] • **MINLPbb** Fletcher and Leyffer [1998] • **Minotaur** Mahajan et al. [2017] • **Mosek** • **Muriqui** Melo et al. [2018] • **NOMAD** Le Digabel [2011] • **Octeract** • **Pavito** Coey et al. [2020] • **SBB** GAMS • **SCIP** Vigerske and Gleixner [2018] • **SHOT** Lundell et al. [2018]

How to solve MINLP?[7]



[7] α**BB** Adjiman et al. [1998a] • α**ECP** Westerlund and Pörn [2002] • **ANTIGONE** Misener and Floudas [2014] • **AOA** Hunting [2011] • **Baron** Tawarmalani and Sahinidis [2005] • **Bonmin** Bonami et al. [2008] • **Couenne** Belotti et al. [2009] • **CPLEX** Bonami et al. [2016] • **DICOPT** Viswanathan and Grossmann [1990] • **EAGO** Wilhelm and Stuber [2017] • **FilMINT** Abhishek et al. [2010] • **GALINI** • **Gurobi** • **Juniper** Kröger et al. [2018] • **Knitro** Byrd et al. [2006] • **LaGO** Nowak et al. [2003] • **LINDO** Gau and Schrage [2004] • **MAiNGO** Najman and Mitsos [2019] • **MILANO** Benson [2011] • **MINLPbb** Fletcher and Leyffer [1998] • **Minotaur** Mahajan et al. [2017] • **Mosek** • **Muriqui** Melo et al. [2018] • **NOMAD** Le Digabel [2011] • **Octeract** • **Pavito** Coey et al. [2020] • **SBB** GAMS • **SCIP** Vigerske and Gleixner [2018] • **SHOT** Lundell et al. [2018]

Situating mixed-integer nonlinear optimization (MINLP)

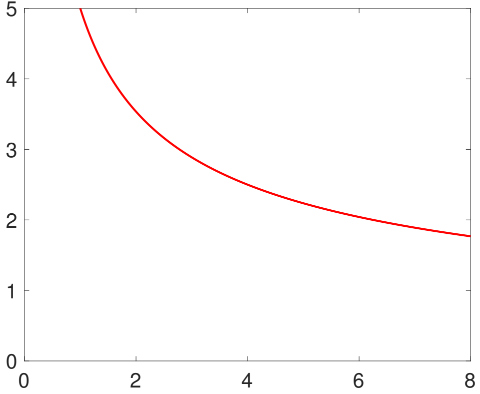
Prerequisites

Mixed-integer linear optimization • Nonlinear optimization • Convexity • Branch & bound

Connections to other work

- **Synomyn** Deterministic global optimization [Floudas and Gounaris, 2009]
- **Subclass** Convex MINLP [Kronqvist et al., 2019]
- **Classification** If computations exact, *complete methods* reach global optimum within a given tolerance in finite time [Neumaier, 2004]
  - *Asymptotically complete* methods reach a global optimum given infinite run time, e.g. Bayesian optimization [Jones et al., 1993]
  - *Rigorous* methods work despite rounding errors [Kearfott, 1996]

What is challenging here?[8]



$$-\sqrt{\frac{25}{x_1}} + x_2 \leq 0$$

$$-1 \cdot x_1^3 + x_2 \leq 0$$

$$(x_1 - 7)^3 + x_2 \leq 0$$

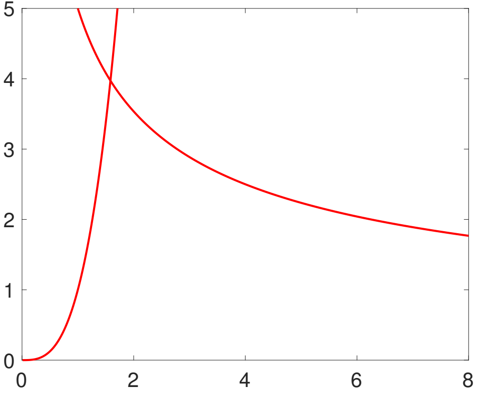
$$-1 \cdot (x_1 - 4)^2 - x_2 \leq -4$$

Required solver elements

Structure detection • Relaxations • Variable bounds • Branching decisions • Primal heuristics

[8] Androulakis et al. [1995], Sahinidis [1996], Liberti and Pantelides [2003], Belotti [2013], Nguyen et al. [2018]

What is challenging here?[8]



$$-\sqrt{\frac{25}{x_1}} + x_2 \le 0$$

$$-1 \cdot x_1^3 + x_2 \le 0$$

$$(x_1 - 7)^3 + x_2 \le 0$$

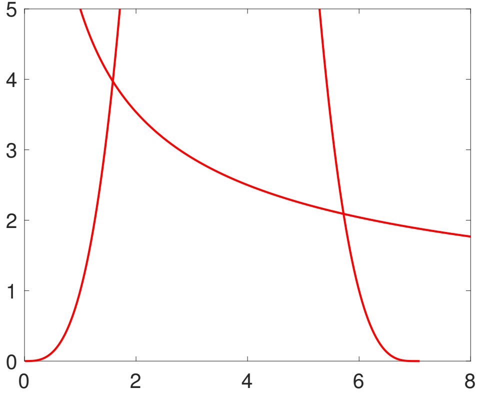
$$-1 \cdot (x_1 - 4)^2 - x_2 \le -4$$

Required solver elements

Structure detection • Relaxations • Variable bounds • Branching decisions • Primal heuristics

[8] Androulakis et al. [1995], Sahinidis [1996], Liberti and Pantelides [2003], Belotti [2013], Nguyen et al. [2018]

What is challenging here?[8]



$$-\sqrt{\frac{25}{x_1}} + x_2 \leq 0$$

$$-1 \cdot x_1^3 + x_2 \leq 0$$

$$(x_1 - 7)^3 + x_2 \leq 0$$

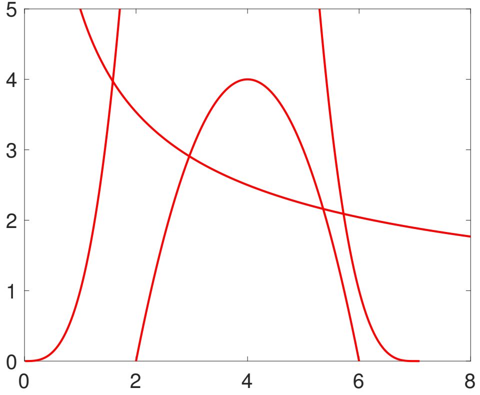
$$-1 \cdot (x_1 - 4)^2 - x_2 \leq -4$$

Required solver elements

Structure detection • Relaxations • Variable bounds • Branching decisions • Primal heuristics

[8] Androulakis et al. [1995], Sahinidis [1996], Liberti and Pantelides [2003], Belotti [2013], Nguyen et al. [2018]

What is challenging here?[8]



$$-\sqrt{\frac{25}{x_1}} + x_2 \leq 0$$

$$-1 \cdot x_1^3 + x_2 \leq 0$$

$$(x_1 - 7)^3 + x_2 \leq 0$$

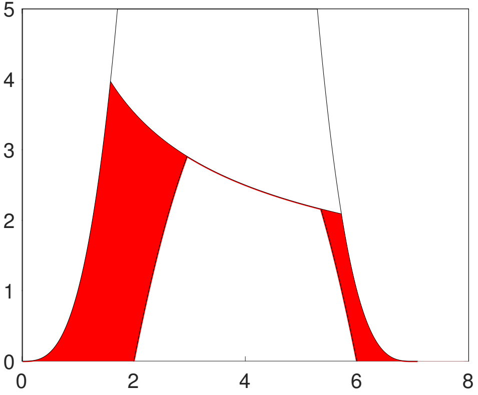
$$-1 \cdot (x_1 - 4)^2 - x_2 \leq -4$$

Required solver elements

Structure detection • Relaxations • Variable bounds • Branching decisions • Primal heuristics

[8] Androulakis et al. [1995], Sahinidis [1996], Liberti and Pantelides [2003], Belotti [2013], Nguyen et al. [2018]

What is challenging here?[8]



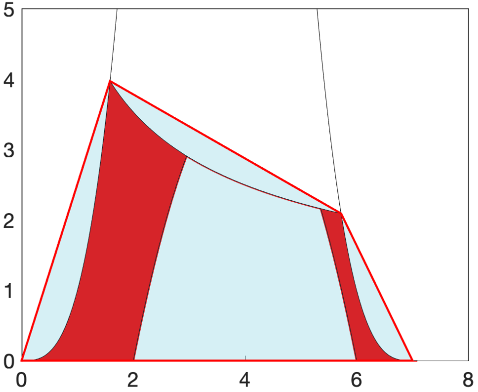
$$
\begin{aligned}
-\sqrt{\frac{25}{x_1}} \quad & +x_2 \leq 0 \\
-1 \cdot x_1^3 \quad & +x_2 \leq 0 \\
(x_1 - 7)^3 \quad & +x_2 \leq 0 \\
-1 \cdot (x_1 - 4)^2 \quad & -x_2 \leq -4
\end{aligned}
$$

Required solver elements

Structure detection • Relaxations • Variable bounds • Branching decisions • Primal heuristics

[8] Androulakis et al. [1995], Sahinidis [1996], Liberti and Pantelides [2003], Belotti [2013], Nguyen et al. [2018]

What is challenging here?[8]



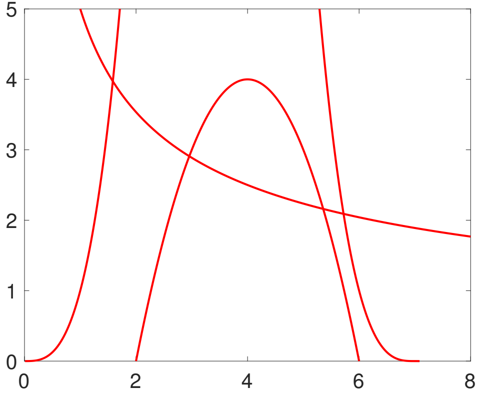
$$
\begin{aligned}
-\sqrt{\frac{25}{x_1}} \quad &+x_2 \leq 0\\
-1 \cdot x_1^3 \quad &+x_2 \leq 0\\
(x_1 - 7)^3 \quad &+x_2 \leq 0\\
-1 \cdot (x_1 - 4)^2 \quad &-x_2 \leq -4
\end{aligned}
$$

Required solver elements

Structure detection • Relaxations • Variable bounds • Branching decisions • Primal heuristics

[8] Androulakis et al. [1995], Sahinidis [1996], Liberti and Pantelides [2003], Belotti [2013], Nguyen et al. [2018]

What is challenging here?[8]



$$-\sqrt{\frac{25}{x_1}} + x_2 \leq 0$$

$$-1 \cdot x_1^3 + x_2 \leq 0$$

$$(x_1 - 7)^3 + x_2 \leq 0$$

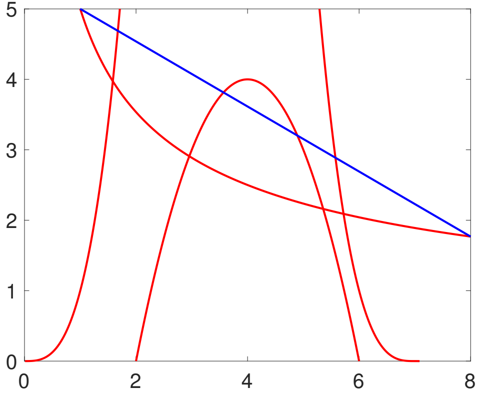
$$-1 \cdot (x_1 - 4)^2 - x_2 \leq -4$$

Required solver elements

Structure detection • Relaxations • Variable bounds • Branching decisions • Primal heuristics

[8] Androulakis et al. [1995], Sahinidis [1996], Liberti and Pantelides [2003], Belotti [2013], Nguyen et al. [2018]

What is challenging here?[8]



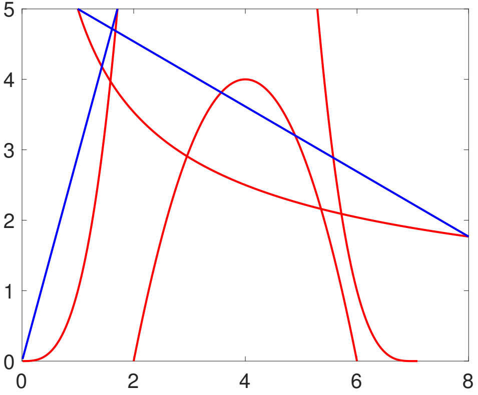
$$
\begin{aligned}
-\sqrt{\frac{25}{x_1}} \quad & +x_2 \le 0 \\
-1 \cdot x_1^3 \quad & +x_2 \le 0 \\
(x_1 - 7)^3 \quad & +x_2 \le 0 \\
-1 \cdot (x_1 - 4)^2 \quad & -x_2 \le -4
\end{aligned}
$$

Required solver elements

Structure detection • Relaxations • Variable bounds • Branching decisions • Primal heuristics

[8] Androulakis et al. [1995], Sahinidis [1996], Liberti and Pantelides [2003], Belotti [2013], Nguyen et al. [2018]

What is challenging here?[8]



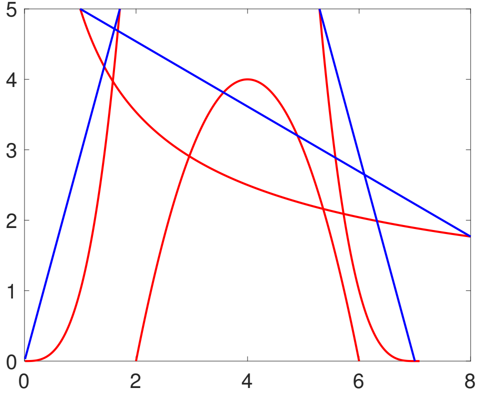
$$
\begin{aligned}
-\sqrt{\frac{25}{x_1}} \quad & +x_2 \leq 0 \\
-1 \cdot x_1^3 \quad & +x_2 \leq 0 \\
(x_1 - 7)^3 \quad & +x_2 \leq 0 \\
-1 \cdot (x_1 - 4)^2 \quad & -x_2 \leq -4
\end{aligned}
$$

Required solver elements

Structure detection • Relaxations • Variable bounds • Branching decisions • Primal heuristics

[8] Androulakis et al. [1995], Sahinidis [1996], Liberti and Pantelides [2003], Belotti [2013], Nguyen et al. [2018]

What is challenging here?[8]



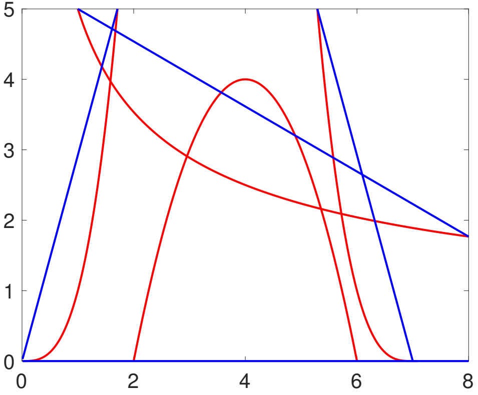
$$
\begin{aligned}
-\sqrt{\frac{25}{x_1}} &\quad +x_2 \leq 0\\
-1 \cdot x_1^3 &\quad +x_2 \leq 0\\
(x_1 - 7)^3 &\quad +x_2 \leq 0\\
-1 \cdot (x_1 - 4)^2 &\quad -x_2 \leq -4
\end{aligned}
$$

Required solver elements

Structure detection • Relaxations • Variable bounds • Branching decisions • Primal heuristics

[8] Androulakis et al. [1995], Sahinidis [1996], Liberti and Pantelides [2003], Belotti [2013], Nguyen et al. [2018]

What is challenging here?[8]



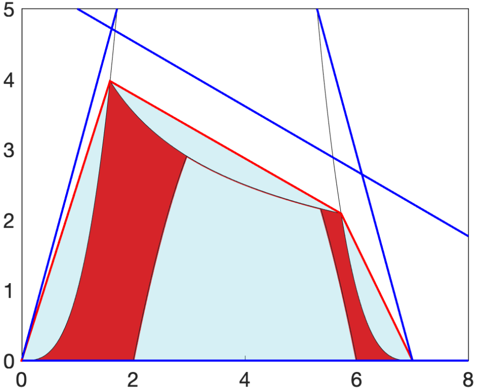
$$
\begin{aligned}
-\sqrt{\frac{25}{x_1}} \quad & +x_2 \leq 0 \\
-1 \cdot x_1^3 \quad & +x_2 \leq 0 \\
(x_1 - 7)^3 \quad & +x_2 \leq 0 \\
-1 \cdot (x_1 - 4)^2 \quad & -x_2 \leq -4
\end{aligned}
$$

Required solver elements

Structure detection • Relaxations • Variable bounds • Branching decisions • Primal heuristics

[8] Androulakis et al. [1995], Sahinidis [1996], Liberti and Pantelides [2003], Belotti [2013], Nguyen et al. [2018]

What is challenging here?[8]



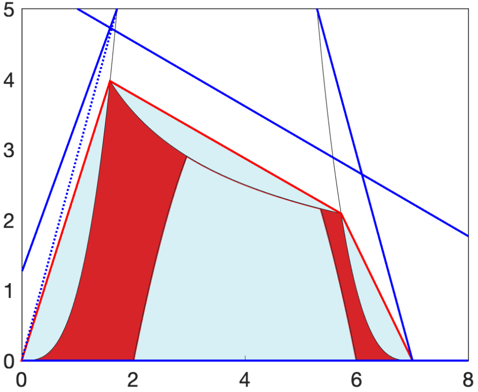
$$
\begin{aligned}
-\sqrt{\frac{25}{x_1}} \quad & +x_2 \leq 0 \\
-1 \cdot x_1^3 \quad & +x_2 \leq 0 \\
(x_1 - 7)^3 \quad & +x_2 \leq 0 \\
-1 \cdot (x_1 - 4)^2 \quad & -x_2 \leq -4
\end{aligned}
$$

Required solver elements

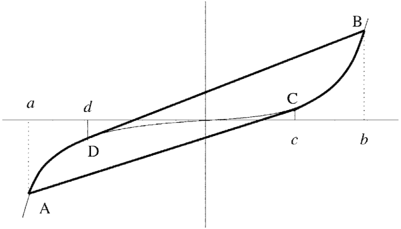
Structure detection • Relaxations • Variable bounds • Branching decisions • Primal heuristics

[8] Androulakis et al. [1995], Sahinidis [1996], Liberti and Pantelides [2003], Belotti [2013], Nguyen et al. [2018]

What is challenging here?[8]



$$
\begin{aligned}
-\sqrt{\frac{25}{x_1}} \quad & +x_2 \leq 0 \\
-1 \cdot x_1^3 \quad & +x_2 \leq 0 \\
(x_1 - 7)^3 \quad & +x_2 \leq 0 \\
-1 \cdot (x_1 - 4)^2 \quad & -x_2 \leq -4
\end{aligned}
$$

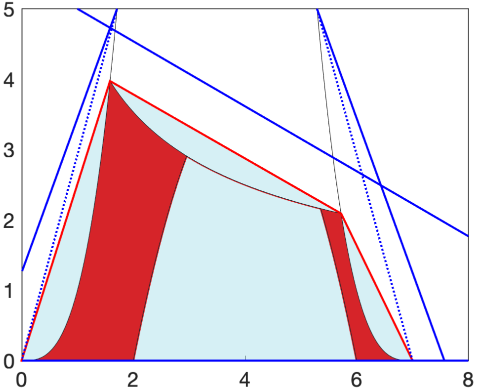


Required solver elements

Structure detection • Relaxations • Variable bounds • Branching decisions • Primal heuristics

[8] Androulakis et al. [1995], Sahinidis [1996], Liberti and Pantelides [2003], Belotti [2013], Nguyen et al. [2018]

What is challenging here?[8]

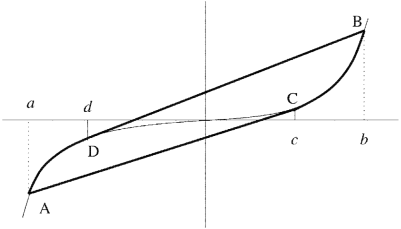


$$-\sqrt{\frac{25}{x_1}} \quad +x_2 \leq 0$$

$$-1 \cdot x_1^3 \quad +x_2 \leq 0$$

$$(x_1 - 7)^3 \quad +x_2 \leq 0$$

$$-1 \cdot (x_1 - 4)^2 \quad -x_2 \leq -4$$

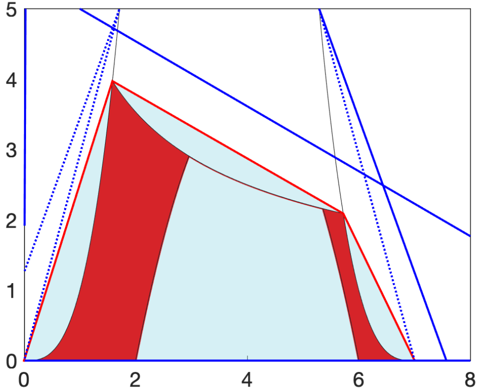


Required solver elements

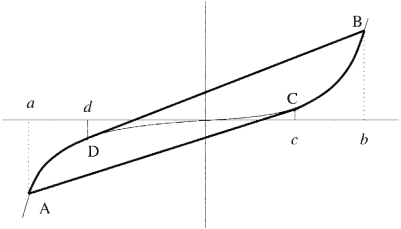
Structure detection • Relaxations • Variable bounds • Branching decisions • Primal heuristics

[8] Androulakis et al. [1995], Sahinidis [1996], Liberti and Pantelides [2003], Belotti [2013], Nguyen et al. [2018]

What is challenging here?[8]



$$
\begin{aligned}
-\sqrt{\frac{25}{x_1}} \quad & +x_2 \leq 0 \\
-1 \cdot x_1^3 \quad & +x_2 \leq 0 \\
(x_1 - 7)^3 \quad & +x_2 \leq 0 \\
-1 \cdot (x_1 - 4)^2 \quad & -x_2 \leq -4
\end{aligned}
$$

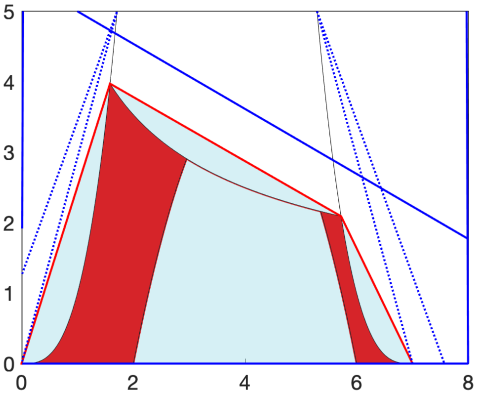


Required solver elements

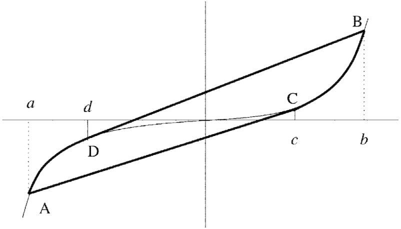
Structure detection • Relaxations • Variable bounds • Branching decisions • Primal heuristics

[8] Androulakis et al. [1995], Sahinidis [1996], Liberti and Pantelides [2003], Belotti [2013], Nguyen et al. [2018]

What is challenging here?[8]



$$
\begin{aligned}
-\sqrt{\frac{25}{x_1}} \quad & +x_2 \le 0 \\
-1 \cdot x_1^3 \quad & +x_2 \le 0 \\
(x_1 - 7)^3 \quad & +x_2 \le 0 \\
-1 \cdot (x_1 - 4)^2 \quad & -x_2 \le -4
\end{aligned}
$$

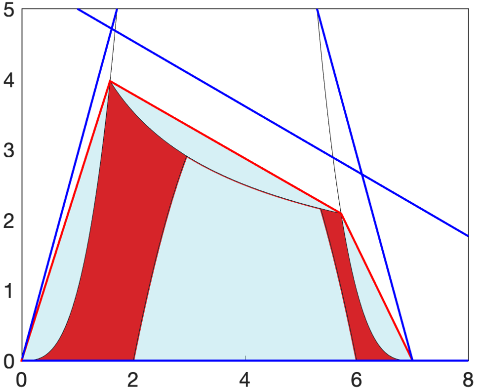


Required solver elements

Structure detection • Relaxations • Variable bounds • Branching decisions • Primal heuristics

[8] Androulakis et al. [1995], Sahinidis [1996], Liberti and Pantelides [2003], Belotti [2013], Nguyen et al. [2018]

What is challenging here?[8]



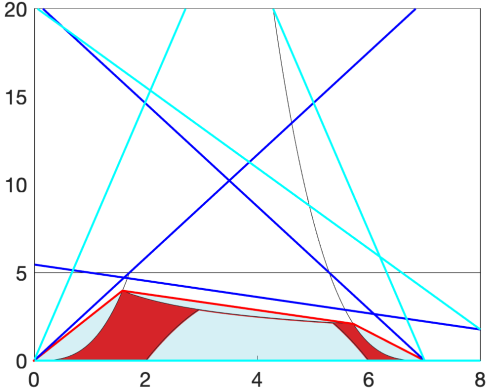
$$
\begin{aligned}
-\sqrt{\frac{25}{x_1}} \qquad & +x_2 \leq 0 \\
-1 \cdot x_1^3 \qquad & +x_2 \leq 0 \\
(x_1 - 7)^3 \qquad & +x_2 \leq 0 \\
-1 \cdot (x_1 - 4)^2 \qquad & -x_2 \leq -4
\end{aligned}
$$

Required solver elements

Structure detection • Relaxations • Variable bounds • Branching decisions • Primal heuristics

[8] Androulakis et al. [1995], Sahinidis [1996], Liberti and Pantelides [2003], Belotti [2013], Nguyen et al. [2018]

What is challenging here?[8]



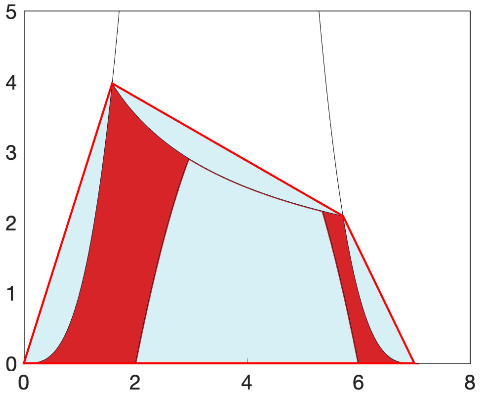
$$
\begin{aligned}
-\sqrt{\frac{25}{x_1}} \quad & +x_2 \leq 0 \\
-1 \cdot x_1^3 \quad & +x_2 \leq 0 \\
(x_1 - 7)^3 \quad & +x_2 \leq 0 \\
-1 \cdot (x_1 - 4)^2 \quad & -x_2 \leq -4
\end{aligned}
$$

Required solver elements

Structure detection • Relaxations • Variable bounds • Branching decisions • Primal heuristics

[8] Androulakis et al. [1995], Sahinidis [1996], Liberti and Pantelides [2003], Belotti [2013], Nguyen et al. [2018]

What is challenging here?[8]



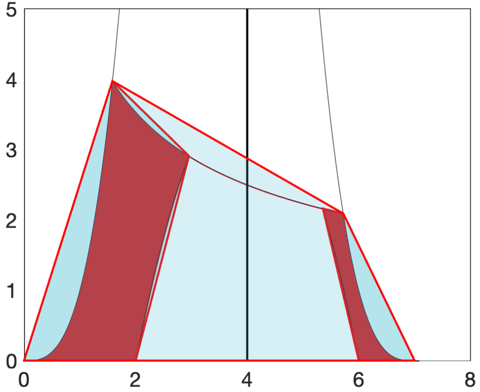
$$
\begin{aligned}
-\sqrt{\frac{25}{x_1}} \quad & +x_2 \leq 0 \\
-1 \cdot x_1^3 \quad & +x_2 \leq 0 \\
(x_1 - 7)^3 \quad & +x_2 \leq 0 \\
-1 \cdot (x_1 - 4)^2 \quad & -x_2 \leq -4
\end{aligned}
$$

Required solver elements

Structure detection • Relaxations • Variable bounds • Branching decisions • Primal heuristics

[8] Androulakis et al. [1995], Sahinidis [1996], Liberti and Pantelides [2003], Belotti [2013], Nguyen et al. [2018]

What is challenging here?[8]



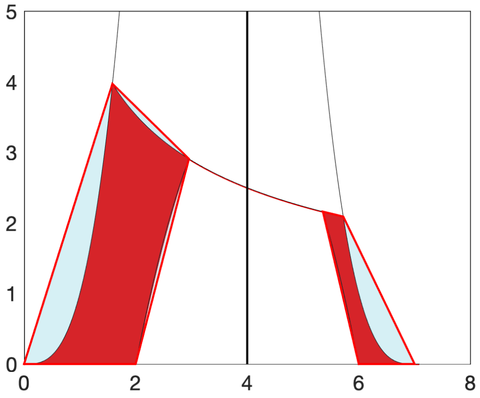
$$
\begin{aligned}
-\sqrt{\frac{25}{x_1}} \quad & +x_2 \le 0 \\
-1 \cdot x_1^3 \quad & +x_2 \le 0 \\
(x_1 - 7)^3 \quad & +x_2 \le 0 \\
-1 \cdot (x_1 - 4)^2 \quad & -x_2 \le -4
\end{aligned}
$$

Required solver elements

Structure detection • Relaxations • Variable bounds • Branching decisions • Primal heuristics

[8] Androulakis et al. [1995], Sahinidis [1996], Liberti and Pantelides [2003], Belotti [2013], Nguyen et al. [2018]

What is challenging here?[8]



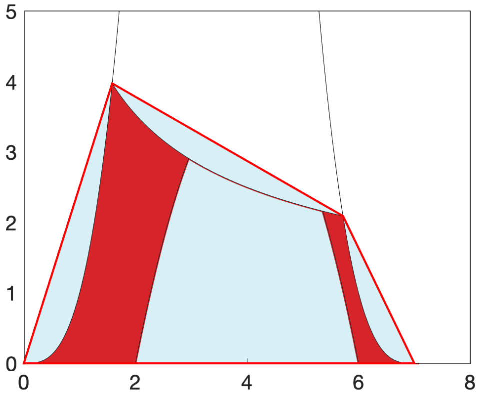
$$
\begin{aligned}
-\sqrt{\frac{25}{x_1}} \quad & +x_2 \le 0 \\
-1 \cdot x_1^3 \quad & +x_2 \le 0 \\
(x_1 - 7)^3 \quad & +x_2 \le 0 \\
-1 \cdot (x_1 - 4)^2 \quad & -x_2 \le -4
\end{aligned}
$$

Required solver elements

Structure detection • Relaxations • Variable bounds • Branching decisions • Primal heuristics

[8] Androulakis et al. [1995], Sahinidis [1996], Liberti and Pantelides [2003], Belotti [2013], Nguyen et al. [2018]

What is challenging here?[8]



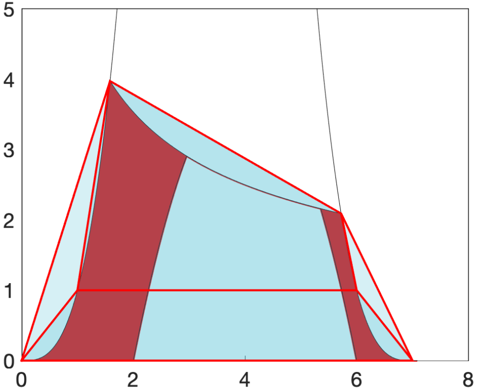
$$
\begin{aligned}
-\sqrt{\frac{25}{x_1}} \quad & +x_2 \leq 0 \\
-1 \cdot x_1^3 \quad & +x_2 \leq 0 \\
(x_1 - 7)^3 \quad & +x_2 \leq 0 \\
-1 \cdot (x_1 - 4)^2 \quad & -x_2 \leq -4
\end{aligned}
$$

Required solver elements

Structure detection • Relaxations • Variable bounds • Branching decisions • Primal heuristics

[8] Androulakis et al. [1995], Sahinidis [1996], Liberti and Pantelides [2003], Belotti [2013], Nguyen et al. [2018]

What is challenging here?[8]



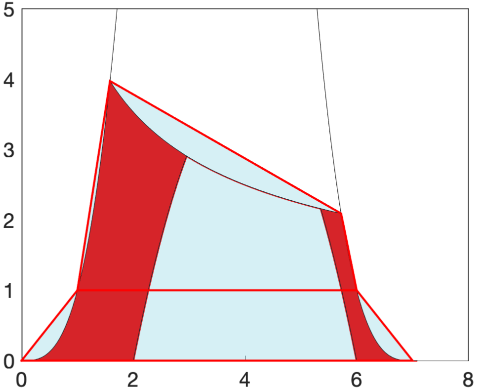
$$
\begin{aligned}
-\sqrt{\frac{25}{x_1}} \quad & +x_2 \leq 0 \\
-1 \cdot x_1^3 \quad & +x_2 \leq 0 \\
(x_1 - 7)^3 \quad & +x_2 \leq 0 \\
-1 \cdot (x_1 - 4)^2 \quad & -x_2 \leq -4
\end{aligned}
$$

Required solver elements

Structure detection • Relaxations • Variable bounds • Branching decisions • Primal heuristics

[8] Androulakis et al. [1995], Sahinidis [1996], Liberti and Pantelides [2003], Belotti [2013], Nguyen et al. [2018]

What is challenging here?[8]



$$-\sqrt{\frac{25}{x_1}} + x_2 \le 0$$

$$-1 \cdot x_1^3 + x_2 \le 0$$

$$(x_1 - 7)^3 + x_2 \le 0$$

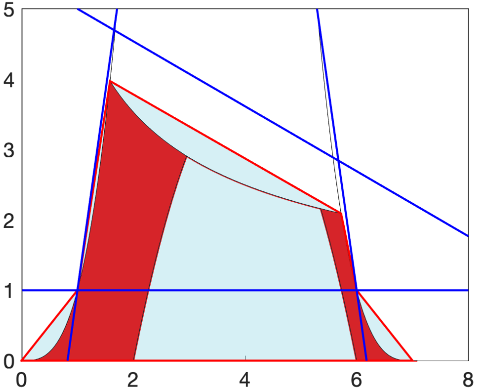
$$-1 \cdot (x_1 - 4)^2 - x_2 \le -4$$

Required solver elements

Structure detection • Relaxations • Variable bounds • Branching decisions • Primal heuristics

[8] Androulakis et al. [1995], Sahinidis [1996], Liberti and Pantelides [2003], Belotti [2013], Nguyen et al. [2018]

What is challenging here?[8]



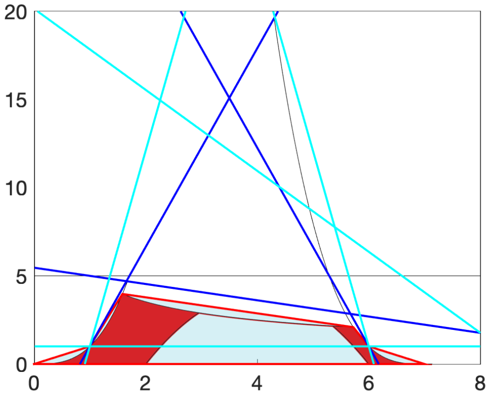
$$
\begin{aligned}
-\sqrt{\frac{25}{x_1}} \quad & +x_2 \leq 0 \\
-1 \cdot x_1^3 \quad & +x_2 \leq 0 \\
(x_1 - 7)^3 \quad & +x_2 \leq 0 \\
-1 \cdot (x_1 - 4)^2 \quad & -x_2 \leq -4
\end{aligned}
$$

Required solver elements

Structure detection • Relaxations • Variable bounds • Branching decisions • Primal heuristics

[8] Androulakis et al. [1995], Sahinidis [1996], Liberti and Pantelides [2003], Belotti [2013], Nguyen et al. [2018]

What is challenging here?[8]



$$-\sqrt{\frac{25}{x_1}} \quad +x_2 \leq 0$$

$$-1 \cdot x_1^3 \quad +x_2 \leq 0$$

$$(x_1 - 7)^3 \quad +x_2 \leq 0$$

$$-1 \cdot (x_1 - 4)^2 \quad -x_2 \leq -4$$

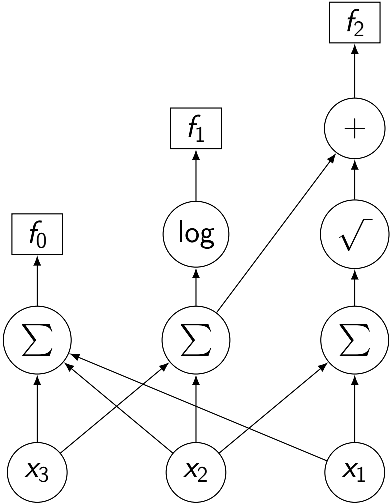
Required solver elements

Structure detection • Relaxations • Variable bounds • Branching decisions • Primal heuristics

[8] Androulakis et al. [1995], Sahinidis [1996], Liberti and Pantelides [2003], Belotti [2013], Nguyen et al. [2018]

Outline

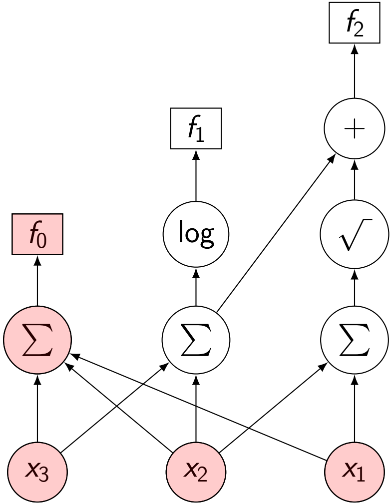
Data structures: Directed acyclic graph[9]



$$
\begin{aligned}
\min_{x_1,x_2,x_3} \quad & x_1 + x_2 + x_3 \\
\text{s.t.} \quad & \log(x_2 - x_3) \geq 0 \\
& x_2 - x_3 + \sqrt{x_1 + x_2} \leq 2 \\
& \mathbf{x} \in \left[\mathbf{x}^{\mathrm{L}}, \mathbf{x}^{\mathrm{U}}\right]
\end{aligned}
$$

[9] Smith and Pantelides [1999], Belotti et al. [2009], Liberti et al. [2010], Liberti [2012], Misener and Floudas [2014], Vigerske and Gleixner [2018], Ceccon et al. [2020]

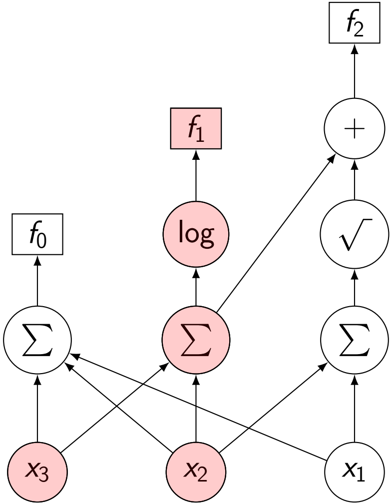
Data structures: Directed acyclic graph[9]



$$\begin{aligned} \min_{x_1,x_2,x_3} \quad & \mathbf{x_1} + \mathbf{x_2} + \mathbf{x_3} \\ \text{s.t.} \quad & \log(x_2 - x_3) \geq 0 \\ & x_2 - x_3 + \sqrt{x_1 + x_2} \leq 2 \\ & \mathbf{x} \in \left[\mathbf{x}^{\mathrm{L}}, \mathbf{x}^{\mathrm{U}}\right] \end{aligned}$$

[9]Smith and Pantelides [1999], Belotti et al. [2009], Liberti et al. [2010], Liberti [2012], Misener and Floudas [2014], Vigerske and Gleixner [2018], Ceccon et al. [2020]

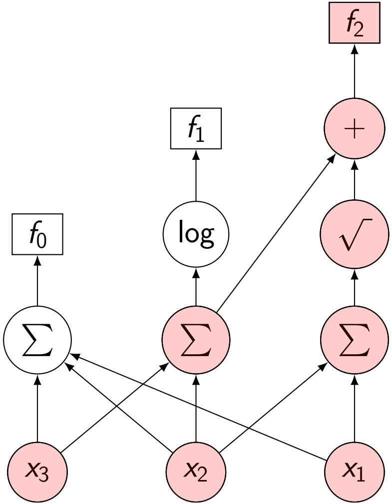
Data structures: Directed acyclic graph[9]



$$\begin{aligned} \min_{x_1,x_2,x_3} \quad & x_1 + x_2 + x_3 \\ \text{s.t.} \quad & \log(\mathbf{x_2} - \mathbf{x_3}) \geq \mathbf{0} \\ & x_2 - x_3 + \sqrt{x_1 + x_2} \leq 2 \\ & \mathbf{x} \in \left[\mathbf{x}^{\mathrm{L}}, \mathbf{x}^{\mathrm{U}}\right] \end{aligned}$$

[9] Smith and Pantelides [1999], Belotti et al. [2009], Liberti et al. [2010], Liberti [2012], Misener and Floudas [2014], Vigerske and Gleixner [2018], Ceccon et al. [2020]

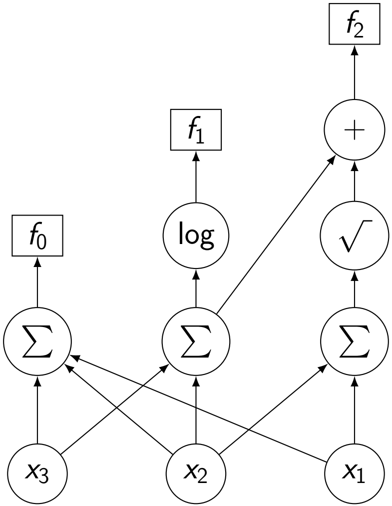
Data structures: Directed acyclic graph[9]

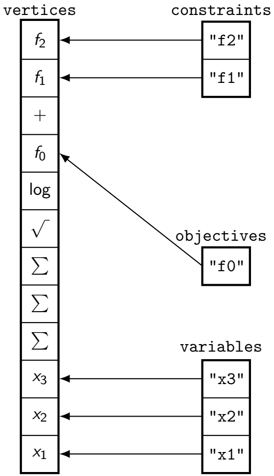


$$\begin{aligned} \min_{x_1,x_2,x_3} \quad & x_1 + x_2 + x_3 \\ \text{s.t.} \quad & \log(x_2 - x_3) \geq 0 \\ & \mathbf{x_2 - x_3 + \sqrt{x_1 + x_2} \leq 2} \\ & \mathbf{x} \in \left[\mathbf{x}^{\mathrm{L}}, \mathbf{x}^{\mathrm{U}}\right] \end{aligned}$$

[9] Smith and Pantelides [1999], Belotti et al. [2009], Liberti et al. [2010], Liberti [2012], Misener and Floudas [2014], Vigerske and Gleixner [2018], Ceccon et al. [2020]

Data structures: Directed acyclic graph[9]

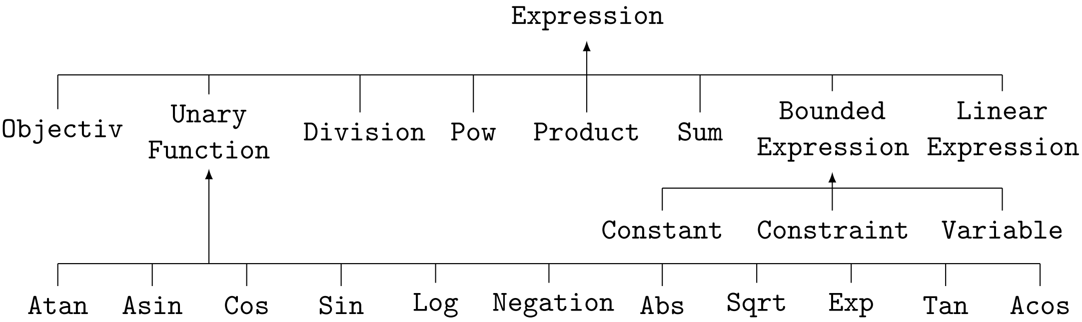




DAG may store vertices in vector, e.g. by depth.

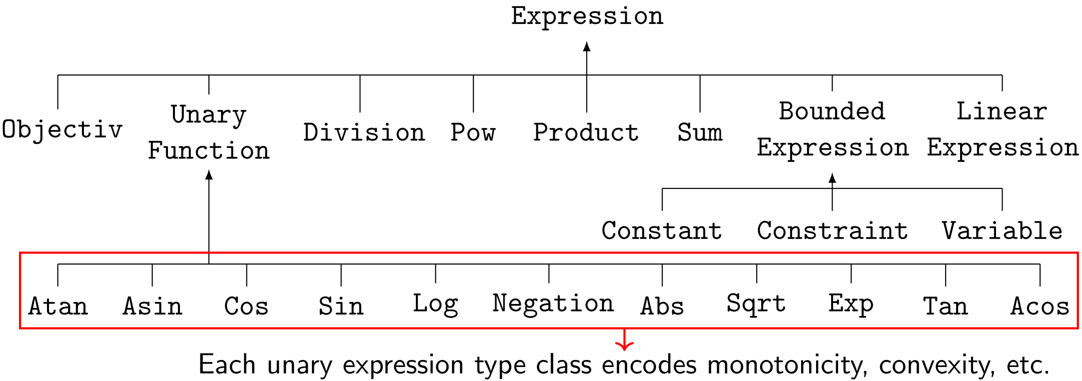
[9] Smith and Pantelides [1999], Belotti et al. [2009], Liberti et al. [2010], Liberti [2012], Misener and Floudas [2014], Vigerske and Gleixner [2018], Ceccon et al. [2020]

Data structures: Hard-coded expression types



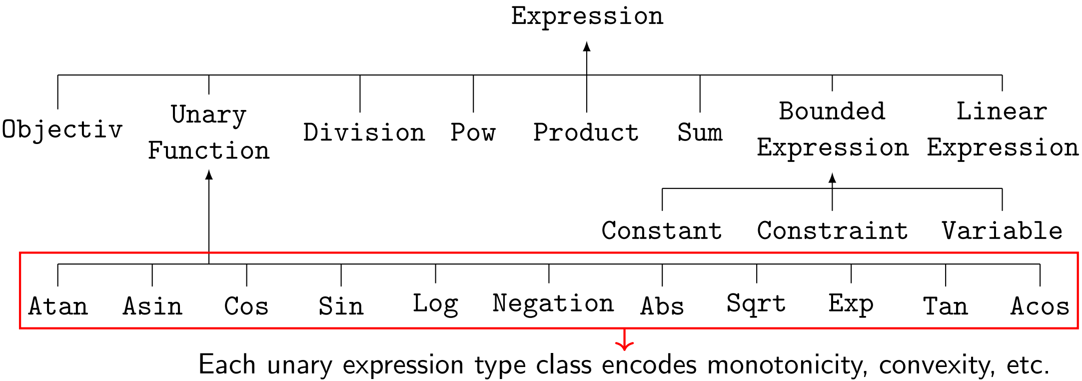
[10] Maranas and Floudas [1997], Ceria and Soares [1999], Lee and Grossmann [2000], Meyer and Floudas [2003, 2004], Frangioni and Gentile [2006], Aktürk et al. [2009], Anstreicher and Burer [2010], Cafieri et al. [2010], Hijazi et al. [2013], Ballerstein and Michaels [2014], Vielma et al. [2017], Speakman and Lee [2017]

Data structures: Hard-coded expression types



[10]Maranas and Floudas [1997], Ceria and Soares [1999], Lee and Grossmann [2000], Meyer and Floudas [2003, 2004], Frangioni and Gentile [2006], Aktürk et al. [2009], Anstreicher and Burer [2010], Cafieri et al. [2010], Hijazi et al. [2013], Ballerstein and Michaels [2014], Vielma et al. [2017], Speakman and Lee [2017]

Data structures: Hard-coded expression types

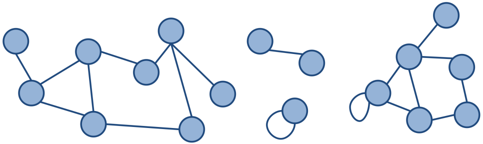


We can hard-code additional rules[10]

Indicator • Second-order cone • Signomial • {Tri-,Quadri-}linear

[10] Maranas and Floudas [1997], Ceria and Soares [1999], Lee and Grossmann [2000], Meyer and Floudas [2003, 2004], Frangioni and Gentile [2006], Aktürk et al. [2009], Anstreicher and Burer [2010], Cafieri et al. [2010], Hijazi et al. [2013], Ballerstein and Michaels [2014], Vielma et al. [2017], Speakman and Lee [2017]

Data structures: Undirected [weighted] graph



$$\min_{\mathbf{x},\mathbf{X}} \quad Q_0 \bullet \mathbf{X} + c_0^T \mathbf{x} + b_0$$

$$Q_k \bullet \mathbf{X} + c_k^T \mathbf{x} \leq b_k \; \forall k \in K$$

$$\mathbf{X} = \mathbf{x}\mathbf{x}^T$$

$$\mathbf{x} \in \left[\mathbf{x}^{\mathrm{L}}, \mathbf{x}^{\mathrm{U}}\right]$$

Quadratic $G(V, E)$

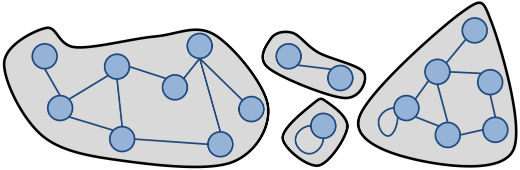
- V Vertex $\implies$ Variable
- E Edge $\implies$ Bilinear term
- [Weight $\implies$ Coefficient]

$$V = \{ \quad 1, 2, \ldots, N\}$$

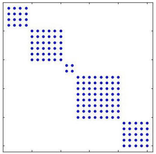
$$E = \{ \quad \{i, j\} \in V \times V \mid i > j, \; \exists k \in K \text{ s.t. } Q_{ij}^{(k)} \neq 0\}$$

$$12 \cdot x_1^2 - 6 \cdot x_1 \cdot x_2 + 6 \cdot x_2^2 - 6.3 \cdot x_3^2 + x_4^2$$

Data structures: Undirected [weighted] graph



$$\min_{\mathbf{x},\mathbf{X}} \quad Q_0 \bullet \mathbf{X} + c_0^T \mathbf{x} + b_0$$
$$Q_k \bullet \mathbf{X} + c_k^T \mathbf{x} \leq b_k \ \forall k \in K$$
$$\mathbf{X} = \mathbf{x}\mathbf{x}^T$$
$$\mathbf{x} \in \left[\mathbf{x}^{\mathrm{L}}, \mathbf{x}^{\mathrm{U}}\right]$$



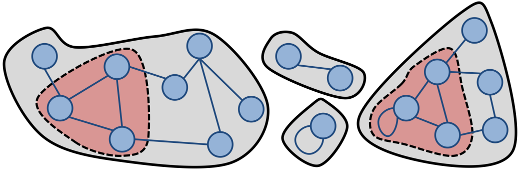
Quadratic $G(V, E)$

- V Vertex $\Longrightarrow$ Variable
- E Edge $\Longrightarrow$ Bilinear term
- [Weight $\Longrightarrow$ Coefficient]

$$V = \{ \quad 1, 2, \ldots, N\}$$
$$E = \{ \quad \{i, j\} \in V \times V \mid i > j, \ \exists k \in K \text{ s.t. } Q_{ij}^{(k)} \neq 0\}$$

$$\underbrace{12 \cdot x_1^2 - 6 \cdot x_1 \cdot x_2 + 6 \cdot x_2^2}_{MT1} - \underbrace{6.3 \cdot x_3^2}_{MT2} + \underbrace{x_4^2}_{MT3}$$

Data structures: Undirected [weighted] graph



$$\min_{\mathbf{x},\mathbf{X}} \quad Q_0 \bullet \mathbf{X} + c_0^T \mathbf{x} + b_0$$

$$Q_k \bullet \mathbf{X} + c_k^T \mathbf{x} \leq b_k \ \forall k \in K$$
$$\mathbf{X} = \mathbf{x}\mathbf{x}^T$$
$$\mathbf{x} \in \left[\mathbf{x}^{\mathrm{L}}, \mathbf{x}^{\mathrm{U}}\right]$$

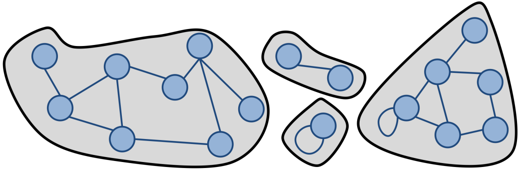
Quadratic $G(V, E)$

- V Vertex $\implies$ Variable
- E Edge $\implies$ Bilinear term
- [Weight $\implies$ Coefficient]

$$V = \{ \quad 1, 2, \ldots, N\}$$
$$E = \{ \quad \{i, j\} \in V \times V \mid i > j, \ \exists k \in K \text{ s.t. } Q_{ij}^{(k)} \neq 0\}$$

[11] Fukuda et al. [2001], Nakata et al. [2003], Baltean-Lugojan et al. [2018], Eltved et al. [2019], [12] Bao et al. [2015], Del Pia and Khajavirad [2018], Del Pia et al. [2018]

Data structures: Undirected [weighted] graph



$$\min_{\mathbf{x},\mathbf{X}} \quad Q_0 \bullet \mathbf{X} + c_0^T \mathbf{x} + b_0$$

$$Q_k \bullet \mathbf{X} + c_k^T \mathbf{x} \leq b_k \ \forall k \in K$$

$$\mathbf{X} = \mathbf{x}\mathbf{x}^T$$

$$\mathbf{x} \in \left[\mathbf{x}^{\mathrm{L}}, \mathbf{x}^{\mathrm{U}}\right]$$

Quadratic $G(V, E)$

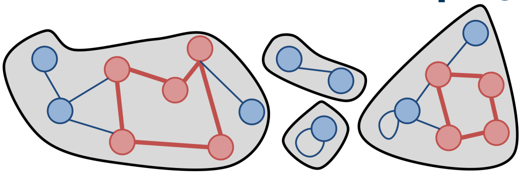
- V Vertex $\implies$ Variable
- E Edge $\implies$ Bilinear term
- [Weight $\implies$ Coefficient]

$$V = \{ \quad 1, 2, \ldots, N\}$$

$$E = \{ \quad \{i, j\} \in V \times V \mid i > j, \ \exists k \in K \text{ s.t. } Q_{ij}^{(k)} \neq 0\}$$

[11] Fukuda et al. [2001], Nakata et al. [2003], Baltean-Lugojan et al. [2018], Eltved et al. [2019], [12] Bao et al. [2015], Del Pia and Khajavirad [2018], Del Pia et al. [2018]

Data structures: Undirected [weighted] graph



$$\min_{\mathbf{x},\mathbf{X}} \quad Q_0 \bullet \mathbf{X} + c_0^T \mathbf{x} + b_0$$
$$Q_k \bullet \mathbf{X} + c_k^T \mathbf{x} \leq b_k \; \forall k \in K$$
$$\mathbf{X} = \mathbf{x}\mathbf{x}^T$$
$$\mathbf{x} \in \left[\mathbf{x}^{\mathrm{L}}, \, \mathbf{x}^{\mathrm{U}}\right]$$

Quadratic $G(V, E)$

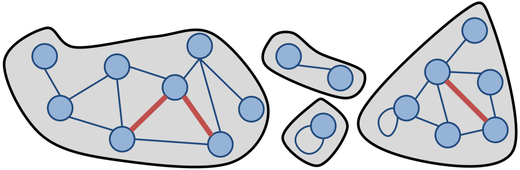
- V Vertex $\implies$ Variable
- E Edge $\implies$ Bilinear term
- [Weight $\implies$ Coefficient]

$$V = \{ \quad 1, 2, \ldots, N\}$$
$$E = \{ \quad \{i, j\} \in V \times V \mid i > j, \; \exists k \in K \text{ s.t. } Q_{ij}^{(k)} \neq 0\}$$

Strengthening with chordal extensions[11]

[11] Fukuda et al. [2001], Nakata et al. [2003], Baltean-Lugojan et al. [2018], Eltved et al. [2019], [12] Bao et al. [2015], Del Pia and Khajavirad [2018], Del Pia et al. [2018]

Data structures: Undirected [weighted] graph



$$\min_{\mathbf{x},\mathbf{X}} \quad Q_0 \bullet \mathbf{X} + c_0^T \mathbf{x} + b_0$$

$$Q_k \bullet \mathbf{X} + c_k^T \mathbf{x} \le b_k \ \forall k \in K$$

$$\mathbf{X} = \mathbf{x}\mathbf{x}^T$$

$$\mathbf{x} \in \left[\mathbf{x}^{\mathrm{L}}, \mathbf{x}^{\mathrm{U}}\right]$$

Quadratic $G(V, E)$

- V Vertex $\Longrightarrow$ Variable
- E Edge $\Longrightarrow$ Bilinear term
- [Weight $\Longrightarrow$ Coefficient]

$$V = \{ \quad 1, 2, \ldots, N\}$$

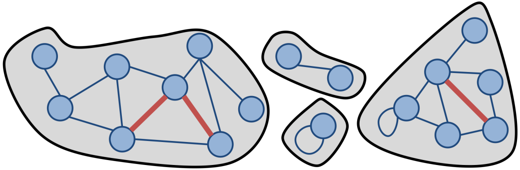
$$E = \{ \quad \{i, j\} \in V \times V \mid i > j, \ \exists k \in K \text{ s.t. } Q_{ij}^{(k)} \neq 0\}$$

Strengthening with chordal extensions[11]

---

[11] Fukuda et al. [2001], Nakata et al. [2003], Baltean-Lugojan et al. [2018], Eltved et al. [2019], [12] Bao et al. [2015], Del Pia and Khajavirad [2018], Del Pia et al. [2018]

Data structures: Undirected [weighted] graph



$$\min_{\mathbf{x},\mathbf{X}} \quad Q_0 \bullet \mathbf{X} + c_0^T \mathbf{x} + b_0$$
$$Q_k \bullet \mathbf{X} + c_k^T \mathbf{x} \le b_k \ \forall k \in K$$
$$\mathbf{X} = \mathbf{x}\mathbf{x}^T$$
$$\mathbf{x} \in \left[\mathbf{x}^{\mathrm{L}}, \mathbf{x}^{\mathrm{U}}\right]$$

Quadratic $G(V, E)$

- V Vertex $\Longrightarrow$ Variable
- E Edge $\Longrightarrow$ Bilinear term
- [Weight $\Longrightarrow$ Coefficient]

$$\begin{aligned} V &= \{ \quad 1, 2, \ldots, N\} \\ E &= \{ \quad \{i, j\} \in V \times V \mid i > j, \\ & \qquad \exists k \in K \text{ s.t. } Q_{ij}^{(k)} \neq 0\} \end{aligned}$$

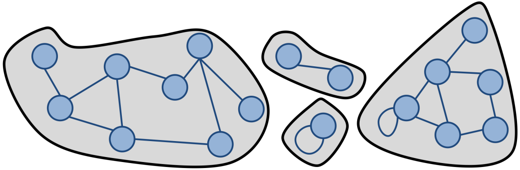
Strengthening with chordal extensions[11]

Extension to multilinear polynomials[12]

Use a hypergraph to allow an edge to join any number of vertices

[11]Fukuda et al. [2001], Nakata et al. [2003], Baltean-Lugojan et al. [2018], Eltved et al. [2019], [12]Bao et al. [2015], Del Pia and Khajavirad [2018], Del Pia et al. [2018]

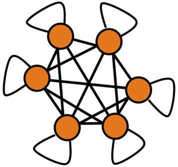
Data structures: Undirected [weighted] graph[13]



Quadratic $G(V, E)$

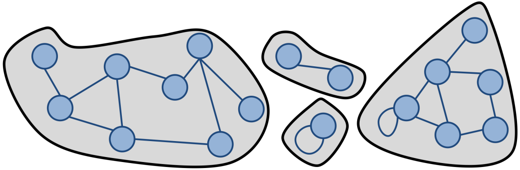
- V Vertex $\Longrightarrow$ Variable
- E Edge $\Longrightarrow$ Bilinear term
- [Weight $\Longrightarrow$ Coefficient]

Common patterns



[13] Misener and Floudas [2012, 2013], Misener et al. [2015]

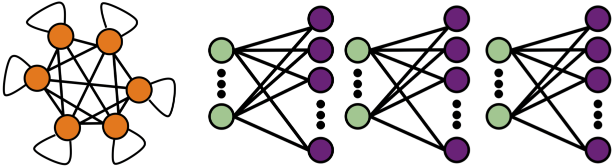
Data structures: Undirected [weighted] graph[13]



Quadratic $G(V, E)$

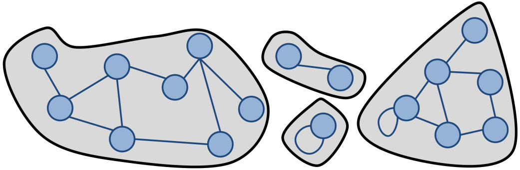
- V Vertex $\implies$ Variable
- E Edge $\implies$ Bilinear term
- [Weight $\implies$ Coefficient]

Common patterns



[13] Misener and Floudas [2012, 2013], Misener et al. [2015]

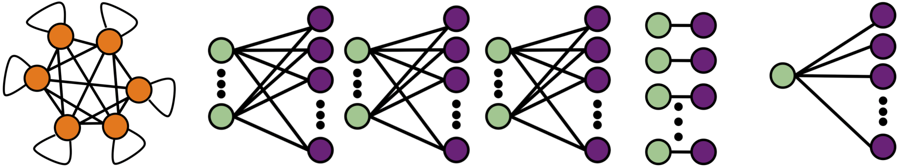
Data structures: Undirected [weighted] graph[13]



Quadratic $G(V, E)$

- V Vertex $\implies$ Variable
- E Edge $\implies$ Bilinear term
- [Weight $\implies$ Coefficient]

Common patterns



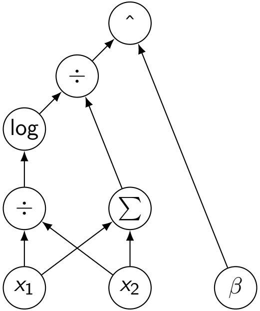
[13] Misener and Floudas [2012, 2013], Misener et al. [2015]

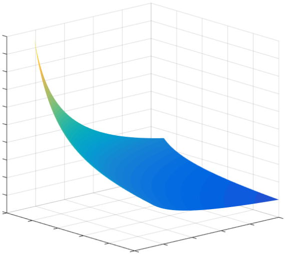
Outline

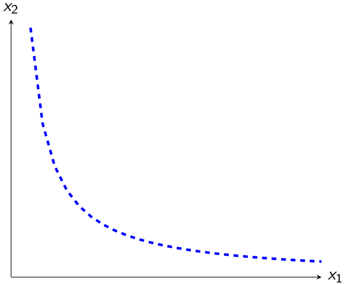
Convex substructures in heat recovery networks

Logarithmic mean temperature difference concave if $-1 \leq \beta \leq 0$ and convex if $\beta \geq 0$ [14]:

$$\text{RecLMTD}^{\beta}(x_1, x_2) = \begin{cases} \left(\frac{\ln(x_1/x_2)}{x_1 - x_2}\right)^{\beta} & x_1 \neq x_2, \\ 1/x_1^{\beta} & x_1 = x_2, \end{cases} \qquad x_1, x_2 \in \mathbb{R}_+, \beta \geq -1.$$





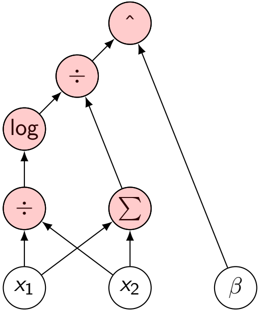


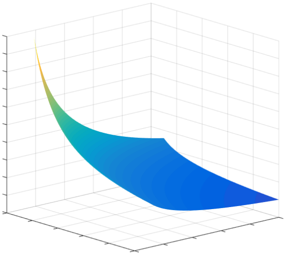
[14] Papalexandri and Pistikopoulos [1994], Mistry and Misener [2016]

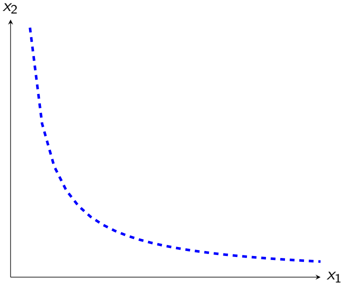
Convex substructures in heat recovery networks

Logarithmic mean temperature difference concave if $-1 \leq \beta \leq 0$ and convex if $\beta \geq 0$ [14]:

$$\mathrm{RecLMTD}^{\beta}(x_1, x_2) = \begin{cases} \left(\frac{\ln(x_1/x_2)}{x_1 - x_2}\right)^{\beta} & x_1 \neq x_2, \\ 1/x_1^{\beta} & x_1 = x_2, \end{cases} \qquad x_1, x_2 \in \mathbb{R}_+, \beta \geq -1.$$





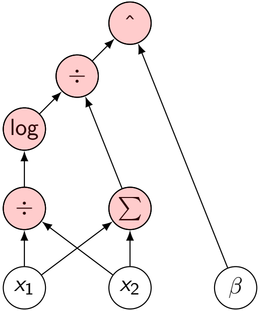


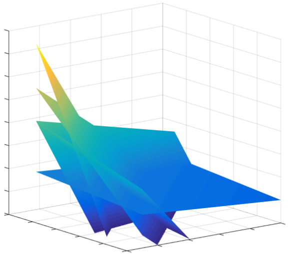
[14] Papalexandri and Pistikopoulos [1994], Mistry and Misener [2016]

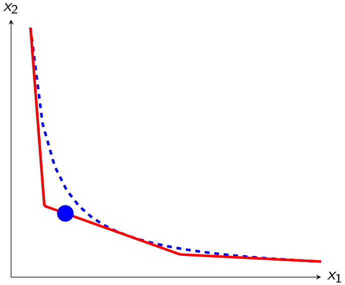
Convex substructures in heat recovery networks

Logarithmic mean temperature difference concave if $-1 \leq \beta \leq 0$ and convex if $\beta \geq 0$ [14]:

$$\mathrm{RecLMTD}^{\beta}(x_1, x_2) = \begin{cases} \left(\frac{\ln(x_1/x_2)}{x_1 - x_2}\right)^{\beta} & x_1 \neq x_2, \\ 1/x_1^{\beta} & x_1 = x_2, \end{cases} \qquad x_1, x_2 \in \mathbb{R}_+, \beta \geq -1.$$





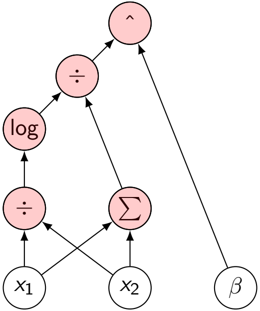


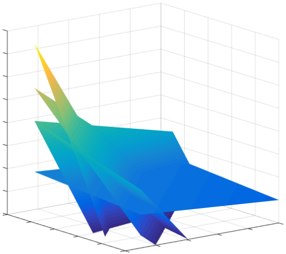
[14] Papalexandri and Pistikopoulos [1994], Mistry and Misener [2016]

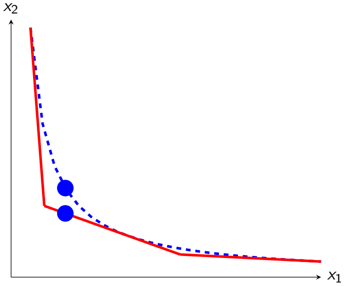
Convex substructures in heat recovery networks

Logarithmic mean temperature difference concave if $-1 \leq \beta \leq 0$ and convex if $\beta \geq 0$ [14]:

$$\text{RecLMTD}^{\beta}(x_1, x_2) = \begin{cases} \left(\frac{\ln(x_1/x_2)}{x_1 - x_2}\right)^{\beta} & x_1 \neq x_2, \\ 1/x_1^{\beta} & x_1 = x_2, \end{cases} \qquad x_1, x_2 \in \mathbb{R}_+, \beta \geq -1.$$





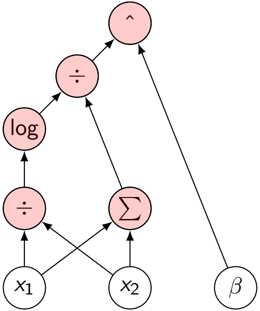


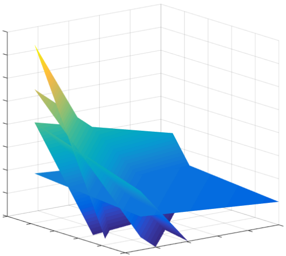
[14] Papalexandri and Pistikopoulos [1994], Mistry and Misener [2016]

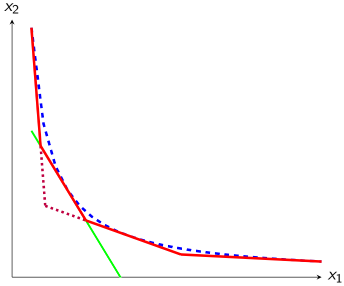
Convex substructures in heat recovery networks

Logarithmic mean temperature difference concave if $-1 \leq \beta \leq 0$ and convex if $\beta \geq 0$ [14]:

$$\text{RecLMTD}^{\beta}(x_1, x_2) = \begin{cases} \left(\frac{\ln(x_1/x_2)}{x_1 - x_2} \right)^{\beta} & x_1 \neq x_2, \\ 1/x_1^{\beta} & x_1 = x_2, \end{cases} \qquad x_1, x_2 \in \mathbb{R}_+, \beta \geq -1.$$







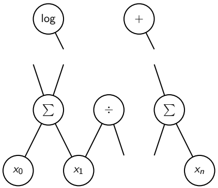
[14] Papalexandri and Pistikopoulos [1994], Mistry and Misener [2016]

Automatic convexity detection

Tree-walking rules [Fourer et al., 2010]

Recursively apply convexity rules to expression trees:

$$k = g_1 \circ g_2, g_1 \text{ convex \& nondecreasing}, g_2 \text{ convex} \implies k \text{ convex}$$



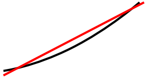
$$\begin{bmatrix} \frac{\partial^2 f}{\partial x_0^2} & \cdots & \frac{\partial^2 f}{\partial x_0, \partial x_n} \\ \vdots & & \vdots \\ \frac{\partial^2 f}{\partial x_n, \partial x_0} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix}$$

Hessian computation [Nenov et al., 2004]

Compute *Hessian sign* with Automatic Differentiation

Sampling [Chinneck, 2001]

Test convexity property by empirical sampling

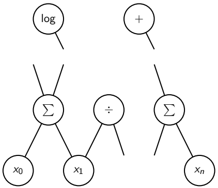


Automatic convexity detection

Tree-walking rules [Fourer et al., 2010]

Recursively apply convexity rules to expression trees:

$$k = g_1 \circ g_2, g_1 \text{ convex \& nondecreasing}, g_2 \text{ convex} \implies k \text{ convex}$$



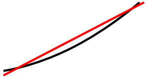
$$\begin{bmatrix} \frac{\partial^2 f}{\partial x_0^2} & \cdots & \frac{\partial^2 f}{\partial x_0, \partial x_n} \\ \vdots & & \vdots \\ \frac{\partial^2 f}{\partial x_n, \partial x_0} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix}$$

Hessian computation [Nenov et al., 2004]

Compute *Hessian sign* with Automatic Differentiation

Sampling [Chinneck, 2001]

Test convexity property by empirical sampling

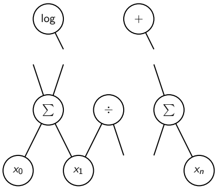


Automatic convexity detection

Tree-walking rules [Fourer et al., 2010]

Recursively apply convexity rules to expression trees:

$k = g_1 \circ g_2, g_1$ convex & nondecreasing, g_2 convex $\implies$ k convex



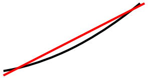
$$\begin{bmatrix} \frac{\partial^2 f}{\partial x_0^2} & \cdots & \frac{\partial^2 f}{\partial x_0, \partial x_n} \\ \vdots & & \vdots \\ \frac{\partial^2 f}{\partial x_n, \partial x_0} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix}$$

Hessian computation [Nenov et al., 2004]

Compute *Hessian sign* with Automatic Differentiation

Sampling [Chinneck, 2001]

Test convexity property by empirical sampling



Solver software strategies

Assume functions are convex (DICOPT, SHOT) • Tag functions as convex (BARON) • Automatic detection (ANTIGONE, BARON, SCIP)

Automatic convexity detection is brittle

Missing hard-coded rule? Get factorable programming decomposition

$$\mathrm{RecLMTD}^{\beta}(x_1, x_2) = \left(\frac{\ln\left(x_1/x_2\right)}{x_1 - x_2}\right)^{\beta} \xrightarrow[\text{form}]{\text{factored}} \begin{cases} v_1 = x_1/x_2 \\ v_2 = \ln v_1 \\ v_3 = x_1 - x_2 \\ v_4 = v_3/v_2 \\ v_5 = v_4^{\beta} \end{cases}$$

[15] Escobar and Grossmann [2010]

Automatic convexity detection is brittle

Missing hard-coded rule? Get factorable programming decomposition

$$\mathrm{RecLMTD}^{\beta}(x_1, x_2) = \left(\frac{\ln(x_1/x_2)}{x_1 - x_2}\right)^{\beta} \xrightarrow[\text{form}]{\text{factored}} \begin{cases} v_1 = x_1/x_2 \\ v_2 = \ln v_1 \\ v_3 = x_1 - x_2 \\ v_4 = v_3/v_2 \\ v_5 = v_4^{\beta} \end{cases}$$

Hard-coded rules are easily tricked

$$\mathrm{RecLMTD}^{\beta}(x_1, x_2) = \left(\frac{\ln(x_1/x_2)}{x_1 - x_2}\right)^{\beta} = \left(\frac{\ln x_1 - \ln x_2}{x_1 - x_2}\right)^{\beta}$$

[15] Escobar and Grossmann [2010]

Automatic convexity detection is brittle

Missing hard-coded rule? Get factorable programming decomposition

$$\mathrm{RecLMTD}^{\beta}(x_1, x_2) = \left(\frac{\ln\left(x_1/x_2\right)}{x_1 - x_2}\right)^{\beta} \xrightarrow[\text{form}]{\text{factored}} \begin{cases} v_1 = x_1/x_2 \\ v_2 = \ln v_1 \\ v_3 = x_1 - x_2 \\ v_4 = v_3/v_2 \\ v_5 = v_4^{\beta} \end{cases}$$

Hard-coded rules are easily tricked

$$\mathrm{RecLMTD}^{\beta}(x_1, x_2) = \left(\frac{\ln\left(x_1/x_2\right)}{x_1 - x_2}\right)^{\beta} = \left(\frac{\ln x_1 - \ln x_2}{x_1 - x_2}\right)^{\beta}$$

Legacy code assumes different numerical strategies [15]

$$\mathrm{RecLMTD}^{\beta}(x_1, x_2) \approx \left(\frac{\ln\left(x_1/(x_2 + \epsilon)\right)}{x_1 - x_2}\right)^{\beta}$$

[15] Escobar and Grossmann [2010]

Disciplined convex programming

Problems are *convex by construction* [Grant and Boyd, 2008]

Impose a set of conventions or rules:

- Atom library holds convex functions,
- Recursively apply expression rules,
- Allow a user to add new atoms to the library.

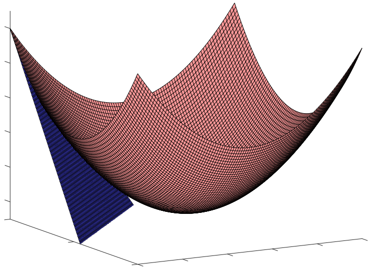
Disciplined convex programming

Problems are *convex by construction* [Grant and Boyd, 2008]

Impose a set of conventions or rules:

- Atom library holds convex functions,
- Recursively apply expression rules,
- Allow a user to add new atoms to the library.

Hybrid: Automatic detection & disciplined programming[16]



We need aspects of both

- Many important functions nonconvex!
- Convex substructures useful, e.g. for extended formulations

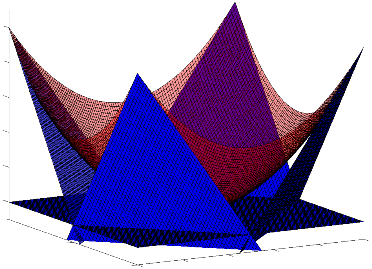
Disciplined convex programming

Problems are *convex by construction* [Grant and Boyd, 2008]

Impose a set of conventions or rules:

- Atom library holds convex functions,
- Recursively apply expression rules,
- Allow a user to add new atoms to the library.

Hybrid: Automatic detection & disciplined programming[16]

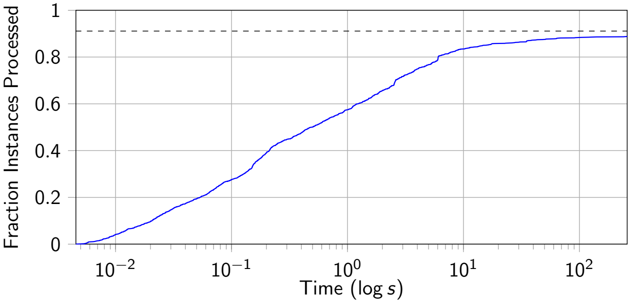


We need aspects of both

- Many important functions nonconvex!
- Convex substructures useful, e.g. for extended formulations

SUSPECT: MINLP special structure detector for Pyomo[17]

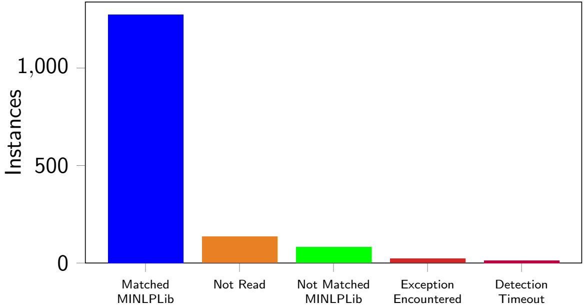
- GitHub: `https://github.com/cog-imperial/suspect`
- SUSPECT extends convexity and monotonicity detection with a plugin-based architecture
- `Setuptools` entry points handle plugin registration and loading
- The default installation includes plugins for:
  - Second-order cone • Perspective function • Fractional expressions



[17] Bussieck et al. [2003], Ceccon et al. [2020]

SUSPECT: MINLP special structure detector for Pyomo[17]

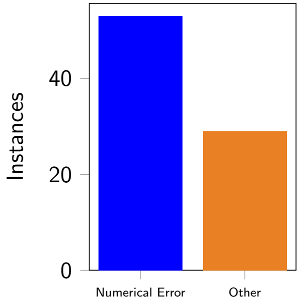
- GitHub: `https://github.com/cog-imperial/suspect`
- SUSPECT extends convexity and monotonicity detection with a plugin-based architecture
- `Setuptools` entry points handle plugin registration and loading
- The default installation includes plugins for:
  - Second-order cone • Perspective function • Fractional expressions



[17] Bussieck et al. [2003], Ceccon et al. [2020]

SUSPECT: MINLP special structure detector for Pyomo[17]

- GitHub: `https://github.com/cog-imperial/suspect`
- SUSPECT extends convexity and monotonicity detection with a plugin-based architecture
- `Setuptools` entry points handle plugin registration and loading
- The default installation includes plugins for:
  - Second-order cone • Perspective function • Fractional expressions

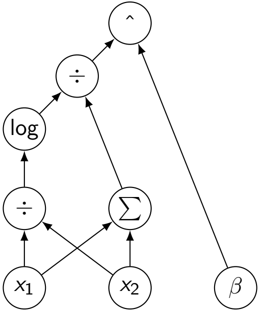


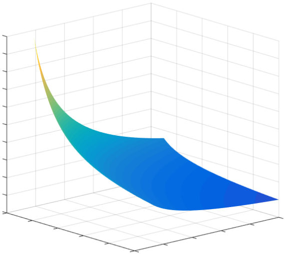
[17] Bussieck et al. [2003], Ceccon et al. [2020]

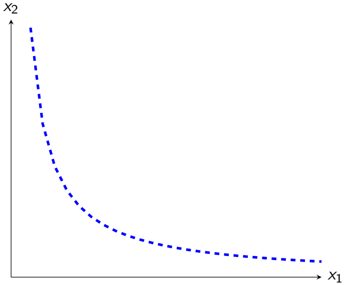
Extending SUSPECT: Example

Logarithmic mean temperature difference concave if $-1 \leq \beta \leq 0$ and convex if $\beta \geq 0$ [18]:

$$\text{RecLMTD}^{\beta}(x_1, x_2) = \begin{cases} \left(\frac{\ln(x_1/x_2)}{x_1 - x_2}\right)^{\beta} & x_1 \neq x_2, \\ 1/x_1^{\beta} & x_1 = x_2, \end{cases} \qquad x_1, x_2 \in \mathbb{R}_+, \beta \geq -1.$$





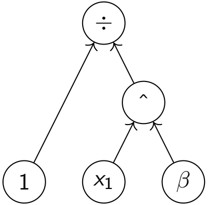


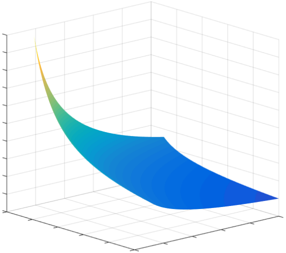
[18] Papalexandri and Pistikopoulos [1994], Mistry and Misener [2016]

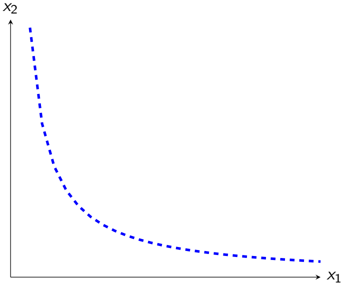
Extending SUSPECT: Example

Logarithmic mean temperature difference concave if $-1 \leq \beta \leq 0$ and convex if $\beta \geq 0$ [18]:

$$\text{RecLMTD}^{\beta}(x_1, x_2) = \begin{cases} \left(\frac{\ln(x_1/x_2)}{x_1 - x_2}\right)^{\beta} & x_1 \neq x_2, \\ 1/x_1^{\beta} & x_1 = x_2, \end{cases} \qquad x_1, x_2 \in \mathbb{R}_+, \beta \geq -1.$$







[18] Papalexandri and Pistikopoulos [1994], Mistry and Misener [2016]

Extending SUSPECT: Example[19]

1. Register the detector using `setuptools` entry points in `setup.py`

```
setup(entry_points={
  'suspect.convexity_detection':
  ['rec_lmtd=rec_lmtd_detector.RecLMTDDetector']
})
```

2. Subclass `suspect.ext.ConvexityDetector` and register rules

```
class RecLMTDDetector(ConvexityDetector):
    def register_rules(self):
        return [
            RecLMTDRule(),
            RecRule(),
        ]
```

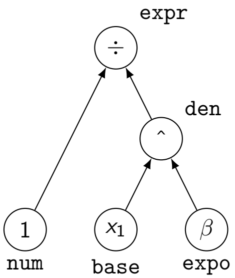
3. Implement rules returning `Convexity` & `Monotonicity` information if known, `None` otherwise

[19] Ceccon et al. [2020]

Extending SUSPECT: Example[20]

```python
class RecRule(Rule):
    root_expr = ExpressionType.Division

    def apply(self, expr, ctx):
        num, den = expr.children
        # numerator has to be 1.0
        if not num.is_constant() or exp.value != 1.0:
            return
        # denominator has to be power with beta >= -1.0
        if not den.expression_type == ExpressionType.Power:
            return
        base, expo = den.children
        if not expo.is_constant() or expo.value < -1.0:
            return
        if not base.expression_type == ExpressionType.Variable:
            return
        bounds = ctx.bounds(base)
        if bounds.is_nonnegative():
            return self._convexity_from_beta(expo.value)
```

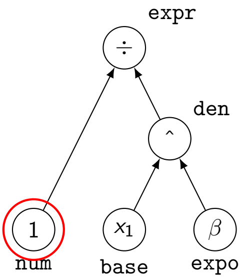


[20] Ceccon et al. [2020]

Extending SUSPECT: Example[20]

```python
class RecRule(Rule):
    root_expr = ExpressionType.Division

    def apply(self, expr, ctx):
        num, den = expr.children
        # numerator has to be 1.0
        if not num.is_constant() or exp.value != 1.0:
            return
        # denominator has to be power with beta >= -1.0
        if not den.expression_type == ExpressionType.Power:
            return
        base, expo = den.children
        if not expo.is_constant() or expo.value < -1.0:
            return
        if not base.expression_type == ExpressionType.Variable:
            return
        bounds = ctx.bounds(base)
        if bounds.is_nonnegative():
            return self._convexity_from_beta(expo.value)
```

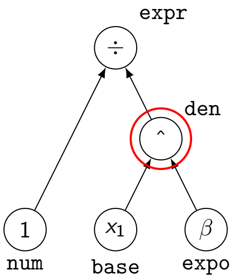


[20] Ceccon et al. [2020]

Extending SUSPECT: Example[20]

```python
class RecRule(Rule):
    root_expr = ExpressionType.Division

    def apply(self, expr, ctx):
        num, den = expr.children
        # numerator has to be 1.0
        if not num.is_constant() or exp.value != 1.0:
            return
        # denominator has to be power with beta >= -1.0
        if not den.expression_type == ExpressionType.Power:
            return
        base, expo = den.children
        if not expo.is_constant() or expo.value < -1.0:
            return
        if not base.expression_type == ExpressionType.Variable:
            return
        bounds = ctx.bounds(base)
        if bounds.is_nonnegative():
            return self._convexity_from_beta(expo.value)
```

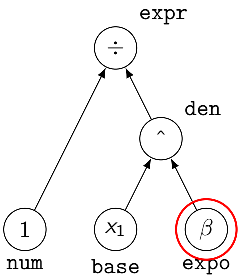


[20] Ceccon et al. [2020]

Extending SUSPECT: Example[20]

```python
class RecRule(Rule):
    root_expr = ExpressionType.Division

    def apply(self, expr, ctx):
        num, den = expr.children
        # numerator has to be 1.0
        if not num.is_constant() or exp.value != 1.0:
            return
        # denominator has to be power with beta >= -1.0
        if not den.expression_type == ExpressionType.Power:
            return
        base, expo = den.children
        if not expo.is_constant() or expo.value < -1.0:
            return
        if not base.expression_type == ExpressionType.Variable:
            return
        bounds = ctx.bounds(base)
        if bounds.is_nonnegative():
            return self._convexity_from_beta(expo.value)
```

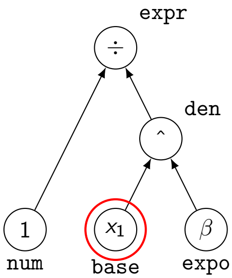


[20]Ceccon et al. [2020]

Extending SUSPECT: Example[20]

```python
class RecRule(Rule):
    root_expr = ExpressionType.Division

    def apply(self, expr, ctx):
        num, den = expr.children
        # numerator has to be 1.0
        if not num.is_constant() or exp.value != 1.0:
            return
        # denominator has to be power with beta >= -1.0
        if not den.expression_type == ExpressionType.Power:
            return
        base, expo = den.children
        if not expo.is_constant() or expo.value < -1.0:
            return
        if not base.expression_type == ExpressionType.Variable:
            return
        bounds = ctx.bounds(base)
        if bounds.is_nonnegative():
            return self._convexity_from_beta(expo.value)
```



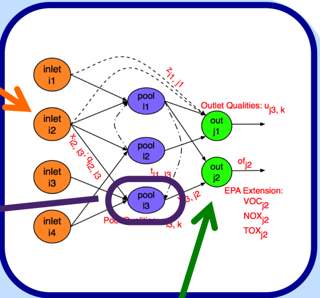
[20] Ceccon et al. [2020]

Structure beyond convexity: Pooling problem[21]

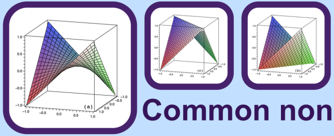
Know input species concentrations (*e.g.*, sulfur)

Variables

$$\sum_{i\in \mathrm{In}} c_{\mathrm{Sulfur},\,i}\cdot f_i = c_{\mathrm{Sulfur,\,Out}}\cdot \sum_{o\in \mathrm{Out}} f_o$$



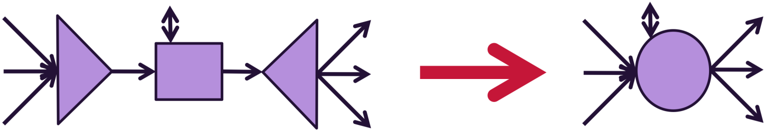
Have output species requirements (e.g., laws)



Common nonlinear term: x × y

[21] **Branching** Dey et al. [2019] • **Relaxations** Adhya et al. [1999], Karuppiah and Grossmann [2006], Meyer and Floudas [2006], Misener et al. [2011], Dey and Gupte [2015], Gupte et al. [2017], Marandi et al. [2017], Dey et al. [2020] • **Cuts** Luedtke et al. [2020] • **Formulations** Alfaki and Haugland [2013], Boland et al. [2016] • **Special structure** Visweswaran and Floudas [1990], Baltean-Lugojan and Misener [2018]

Structure beyond convexity: Local detection[22]



- Product disaggregation for convex relaxations & bounds tightening

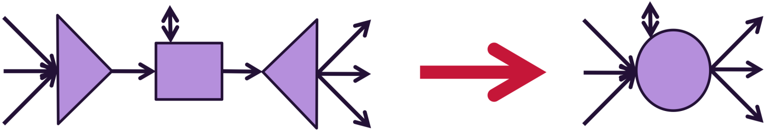
$$\mathbf{p}_{\ell,\mathbf{k}} \cdot \sum_{j \in J} \mathbf{f}_{\ell,\mathbf{j}} = \sum_{j \in J} \mathbf{p}_{\ell,\mathbf{k}} \cdot \mathbf{f}_{\ell,\mathbf{j}}$$

- Find products of linear equality equations & variables

$$\left(\sum_j a_{m,j} \cdot x_j - b_m \right) \cdot \mathbf{x_i} = \sum_j a_{m,j} \cdot x_j \cdot \mathbf{x_i} - b_m \cdot \mathbf{x_i} = 0$$

[22] Adams and Sherali [1990], Tawarmalani and Sahinidis [2002], Liberti and Pantelides [2006], Vigerske et al. [2012], Misener and Floudas [2013], Zorn and Sahinidis [2014], Misener et al. [2015]

Structure beyond convexity: Local detection[22]



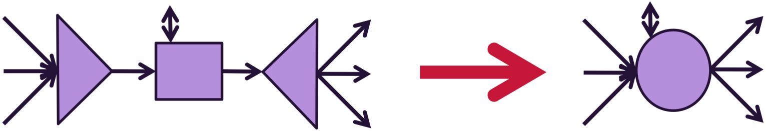
Mixer
$$\begin{cases} \sum_i F_{M,i} = F_T^{IN} \\ \sum_i x_{M,i,j} \cdot F_{M,i} = x_{T,j}^{IN} \cdot F_T^{IN} \quad \forall\, j \in \{1, \ldots, J\} \end{cases}$$

Treatment
$$\begin{cases} F_T^{IN} = F_T^{OUT} \\ x_{T,j}^{IN} = \beta_{T,j} \cdot x_{T,j}^{OUT} \quad \forall\, j \in \{1, \ldots, J\} \end{cases}$$

Splitter
$$\begin{cases} F_T^{OUT} = \sum_i F_{S,i} \\ x_{T,j}^{OUT} = x_{T,i,j} \quad \forall\, i \in \{1, \ldots, I\};\; j \in \{1, \ldots, J\} \end{cases}$$

[22] Adams and Sherali [1990], Tawarmalani and Sahinidis [2002], Liberti and Pantelides [2006], Vigerske et al. [2012], Misener and Floudas [2013], Zorn and Sahinidis [2014], Misener et al. [2015]

Structure beyond convexity: Local detection[22]



Mixer
$$\begin{cases} \sum_i F_{M,i} = F_T^{IN} \\ \sum_i x_{M,i,j} \cdot F_{M,i} = x_{T,j}^{IN} \cdot F_T^{IN} \quad \forall\, j \in \{1, \ldots, J\} \end{cases}$$

Treatment
$$\begin{cases} F_T^{IN} = F_T^{OUT} \\ x_{T,j}^{IN} = \beta_{T,j} \cdot x_{T,j}^{OUT} \quad \forall\, j \in \{1, \ldots, J\} \end{cases}$$

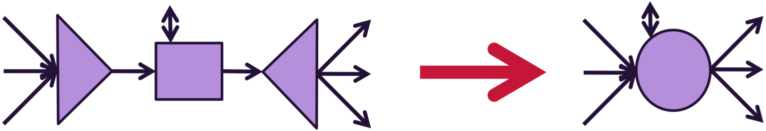
Splitter
$$\begin{cases} F_T^{OUT} = \sum_i F_{S,i} \\ x_{T,j}^{OUT} = x_{T,i,j} \quad \forall\, i \in \{1, \ldots, I\};\ j \in \{1, \ldots, J\} \end{cases}$$

[22] Adams and Sherali [1990], Tawarmalani and Sahinidis [2002], Liberti and Pantelides [2006], Vigerske et al. [2012], Misener and Floudas [2013], Zorn and Sahinidis [2014], Misener et al. [2015]

Structure beyond convexity: Local detection[22]

> 10× Speed-up for 10% of process networks

ANTIGONE, BARON, Couenne, LINDO



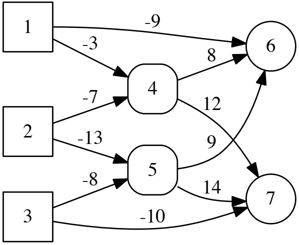
Mixer
$$\begin{cases} \sum_i F_{M,i} = F_T^{IN} \\ \sum_i x_{M,i,j} \cdot F_{M,i} = x_{T,j}^{IN} \cdot F_T^{IN} \quad \forall\, j \in \{1, \ldots, J\} \end{cases}$$

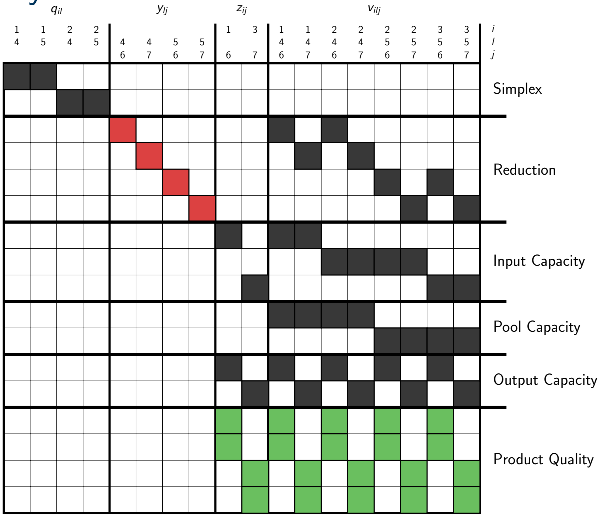
Treatment
$$\begin{cases} F_T^{IN} = F_T^{OUT} \\ x_{T,j}^{IN} = \beta_{T,j} \cdot x_{T,j}^{OUT} \quad \forall\, j \in \{1, \ldots, J\} \end{cases}$$

Splitter
$$\begin{cases} F_T^{OUT} = \sum_i F_{S,i} \\ x_{T,j}^{OUT} = x_{T,i,j} \quad \forall\, i \in \{1, \ldots, I\};\ j \in \{1, \ldots, J\} \end{cases}$$

[22] Adams and Sherali [1990], Tawarmalani and Sahinidis [2002], Liberti and Pantelides [2006], Vigerske et al. [2012], Misener and Floudas [2013], Zorn and Sahinidis [2014], Misener et al. [2015]

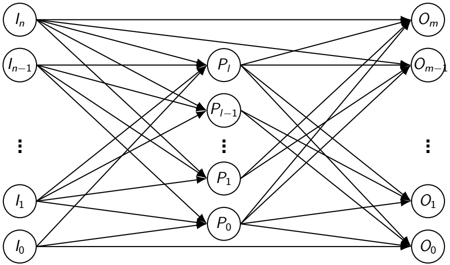
Structure beyond convexity: Global detection[23]





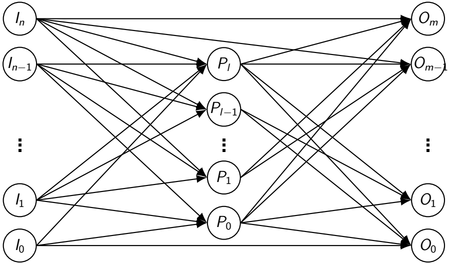
[23] Brown and Wright [1984], Bixby and Fourer [1988], Nemhauser et al. [1994], Achterberg and Raack [2010], Salvagnin [2016], Ceccon et al. [2016]

Structure beyond convexity: Disciplined programming?[24]



[24] Evans et al. [1979], Barton and Pantelides [1994], Ceccon et al. [2020]

Structure beyond convexity: Disciplined programming?[24]

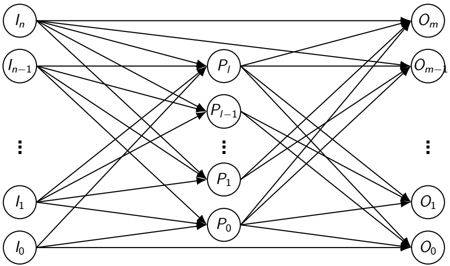


Other networks

- **Gas** [Pfetsch et al., 2015, Humpola and Fügenschuh, 2015, Liers and Merkert, 2016, Bärmann et al., 2018]
- **Power** [Bienstock and Munoz, 2014, Chen et al., 2015, Kocuk et al., 2016]

[24] Evans et al. [1979], Barton and Pantelides [1994], Ceccon et al. [2020]

Structure beyond convexity: Disciplined programming?[24]



Other networks

- **Gas** [Pfetsch et al., 2015, Humpola and Fügenschuh, 2015, Liers and Merkert, 2016, Bärmann et al., 2018]
- **Power** [Bienstock and Munoz, 2014, Chen et al., 2015, Kocuk et al., 2016]

Substructures in other important problems?

Where to go looking for interesting structures ...

- **Sustainability**, e.g. Guillén-Gosálbez and Grossmann [2010]
- **Swing adsorption** with PDEs [Hasan et al., 2012, Leperi et al., 2016]
- **Thermodynamic equilibria** [Bollas et al., 2009, Mitsos et al., 2009a, Pereira et al., 2010, Glass and Mitsos, 2017], e.g. within **distillation** [Nallasivam et al., 2016, Mertens et al., 2018]

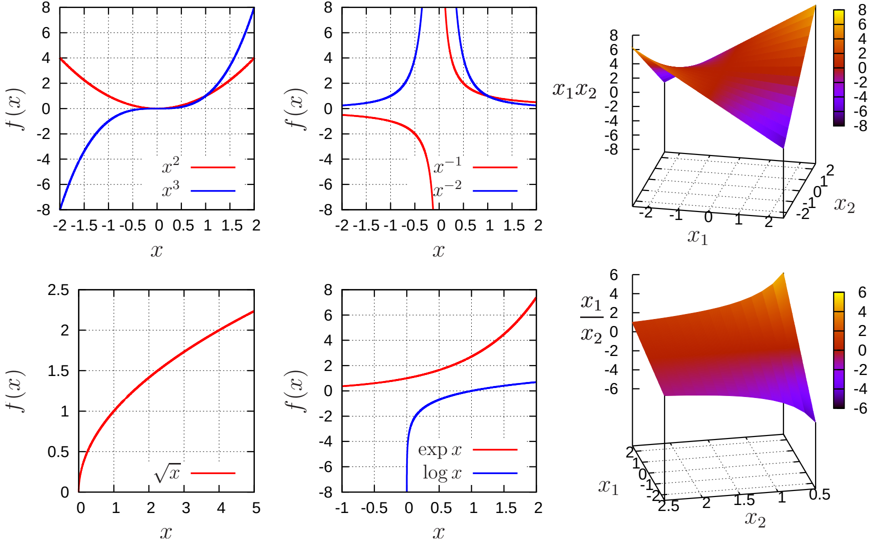
[24] Evans et al. [1979], Barton and Pantelides [1994], Ceccon et al. [2020]

Outline

Outline

Recognize and relax common functions on boxes[25]

Graphics by Benoît Chachuat



Factorable programming: Recursively compose functions

Auxiliary variable method [Smith and Pantelides, 1999]

$$\mathrm{RecLMTD}^{\beta}(x_1, x_2) = \left(\frac{\ln(x_1/x_2)}{x_1 - x_2}\right)^{\beta} \xrightarrow[\text{form}]{\text{factored}} \begin{cases} v_1 = x_1/x_2 \\ v_2 = \ln v_1 \\ v_3 = x_1 - x_2 \\ v_4 = v_3/v_2 \\ v_5 = v_4^{\beta} \end{cases}$$

Factorable programming: Recursively compose functions

Auxiliary variable method [Smith and Pantelides, 1999]

$$\mathrm{RecLMTD}^{\beta}(x_1, x_2) = \left(\frac{\ln(x_1/x_2)}{x_1 - x_2}\right)^{\beta} \xrightarrow[\text{form}]{\text{factored}} \begin{cases} v_1 = x_1/x_2 \\ v_2 = \ln v_1 \\ v_3 = x_1 - x_2 \\ v_4 = v_3/v_2 \\ v_5 = v_4^{\beta} \end{cases}$$

Subgradient propagation method

- **Theory** McCormick [1976], Mitsos et al. [2009b], Tsoukalas and Mitsos [2014], Stuber et al. [2015]
- **Try it out** `EAGO` `https://github.com/PSORLab/EAGO.jl` • `MAiNGO` `https://git.rwth-aachen.de/avt.svt/public/maingo`

Factorable programming: Recursively compose functions

Auxiliary variable method [Smith and Pantelides, 1999]

$$\mathrm{RecLMTD}^{\beta}(x_1, x_2) = \left(\frac{\ln(x_1/x_2)}{x_1 - x_2}\right)^{\beta} \xrightarrow[\text{form}]{\text{factored}} \begin{cases} v_1 = x_1/x_2 \\ v_2 = \ln v_1 \\ v_3 = x_1 - x_2 \\ v_4 = v_3/v_2 \\ v_5 = v_4^{\beta} \end{cases}$$

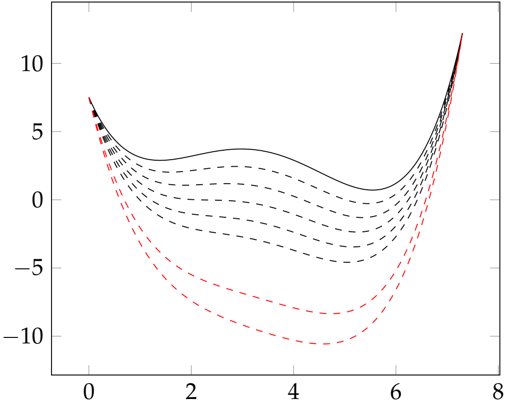
Subgradient propagation method

- **Theory** McCormick [1976], Mitsos et al. [2009b], Tsoukalas and Mitsos [2014], Stuber et al. [2015]
- **Try it out** `EAGO` `https://github.com/PSORLab/EAGO.jl` • `MAiNGO` `https://git.rwth-aachen.de/avt.svt/public/maingo`

Relaxation

`MC++` `https://omega-icl.github.io/mcpp/` • `Coramin` `https://github.com/Coramin/Coramin` , used in `GALINI` `https://github.com/cog-imperial/galini`

αBB: Quadratic shift matrix



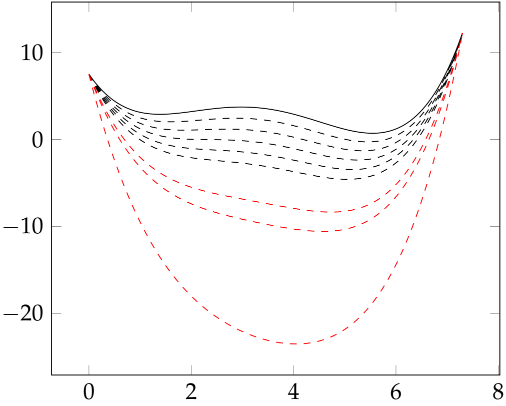
$$f^{\mathsf{CVX}}(\mathbf{x}) = f(\mathbf{x}) + \alpha \left[\mathbf{x} - \mathbf{x}^{\mathrm{L}}\right]^{\mathsf{T}} \left[\mathbf{x} - \mathbf{x}^{\mathrm{U}}\right],$$

$\alpha \implies$ shift parameter

$\mathbf{H}(\mathbf{x}) + 2\alpha\,\mathrm{I} \succeq 0, \ \forall \mathbf{x} \in \mathbf{X}$

Androulakis et al. [1995], Adjiman et al. [1998b,a], Mönnigmann [2011], Schulze Darup and Mönnigmann [2016]

αBB: Quadratic shift matrix



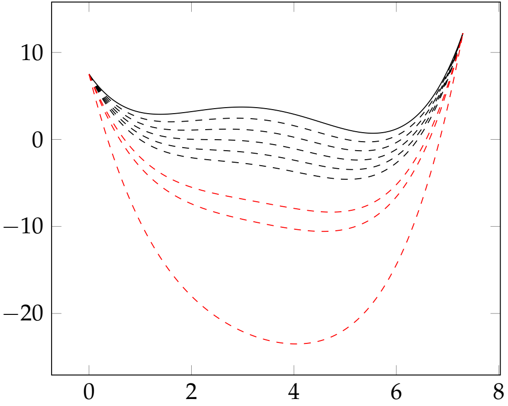
$$f^{\mathsf{CVX}}(\mathbf{x}) = f(\mathbf{x}) + \alpha \left[\mathbf{x} - \mathbf{x}^{\mathrm{L}}\right]^{\mathsf{T}} \left[\mathbf{x} - \mathbf{x}^{\mathrm{U}}\right],$$

$$\alpha \implies \text{shift parameter}$$

$\mathbf{H}(\mathbf{x}) + 2\alpha\,\mathrm{I} \succeq 0, \ \forall \mathbf{x} \in \mathbf{X}$

Androulakis et al. [1995], Adjiman et al. [1998b,a], Mönnigmann [2011], Schulze Darup and Mönnigmann [2016]

αBB: Quadratic shift matrix



$$f^{\text{CVX}}(\mathbf{x}) = f(\mathbf{x}) + \alpha \left[\mathbf{x} - \mathbf{x}^{\mathrm{L}}\right]^{\mathsf{T}} \left[\mathbf{x} - \mathbf{x}^{\mathrm{U}}\right],$$

$$\alpha \implies \text{shift parameter}$$

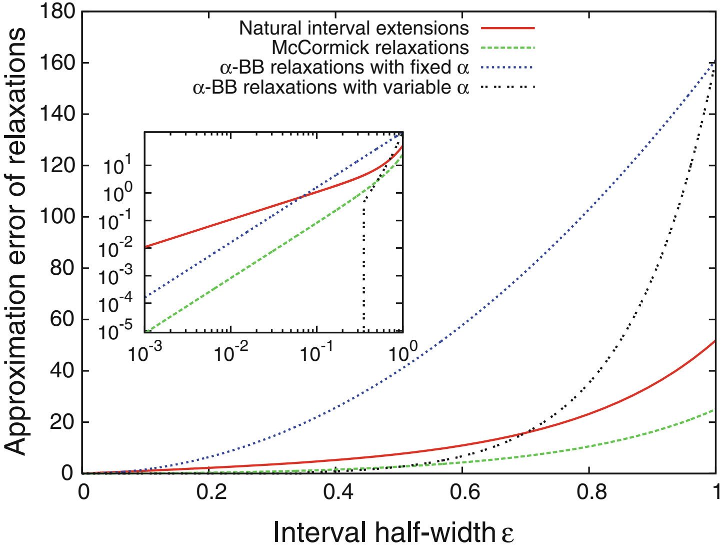
$\mathbf{H}(\mathbf{x}) + 2\alpha\,\mathrm{I} \succeq 0, \ \forall \mathbf{x} \in \mathbf{X}$

Androulakis et al. [1995], Adjiman et al. [1998b,a], Mönnigmann [2011], Schulze Darup and Mönnigmann [2016]

Other relaxation schemes ...

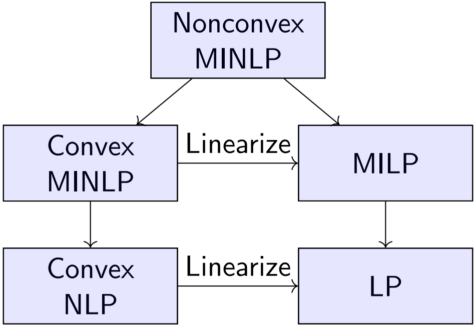
Lasserre [2001] hierarchy for polynomials, e.g. in `GloptiPoly` [Henrion and Lasserre, 2003] • Sum-of-squares and alternatives, e.g. Ahmadi and Majumdar [2019]

Goal: Use all the relaxations at once[26]

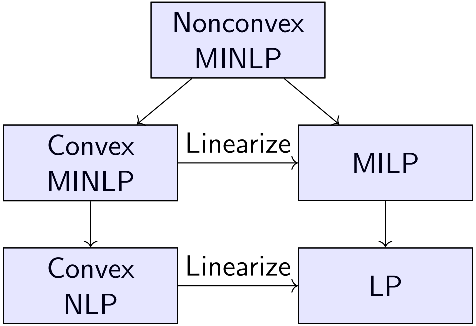


[26] Anstreicher [2009], Bompadre and Mitsos [2012]

Open engineering question: How to relax?



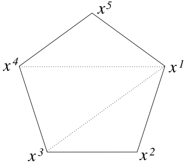
Open engineering question: How to relax?



Open engineering question: Use auxiliary variables?

- **No auxiliary variables** αBB • Factorable programming with subgradient propagation,
- **Use auxiliary variables** Factorable programming with auxiliary variables • SOS-like hierarchies,
- Can I hybridize using auxiliary variables or not?
- Do I have to commit to a strategy at the root node?

Convex relaxations beyond boxes?



Edge concave

Tardella [2004], Meyer and Floudas [2006], Tardella [2008]

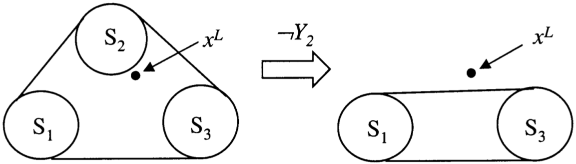
- **Intersect: quadratic & polytope** Linderoth [2005], Tawarmalani et al. [2010], Nguyen et al. [2011], Kocuk et al. [2016], Davarnia et al. [2017], Rahman and Mahajan [2019], Li and Vittal [2018], Santana and Dey [2018], Dey et al. [2019], Anstreicher et al. [2020]
- **Intersect: 2 quadratics** Burer and Kılınç-Karzan [2017], Modaresi and Vielma [2017]
- **Regions** Khajavirad and Sahinidis [2013], Tawarmalani et al. [2013]
- **Intersect: Bivariate function & polytope** Locatelli [2018]

What do we need to integrate into generic solvers?

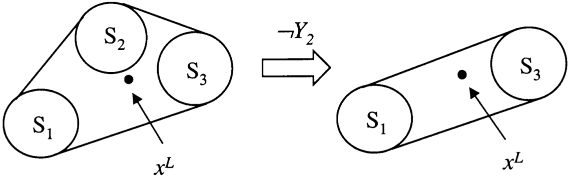
More relaxations • Generic data structures [Kvasnica et al., 2004]

Outline

One branch may not exclude the current relaxation![27]



a) Partitionable set



b) Non-partitionable set

[27] Lee and Grossmann [2000]

How shall we branch?

Integer variables

- Integrate directly into a MIP solver?
  - SCIP [Vigerske and Gleixner, 2018] • CPLEX [Bonami et al., 2016]
- Leave the discrete variables to a state-of-the-art MIP solver?
  - BONMIN [Bonami et al., 2008] • ANTIGONE [Misener and Floudas, 2014]
- Effectively develop a MIP solver together with the MINLP solver?
  - BARON [Tawarmalani and Sahinidis, 2005] • Couenne [Belotti et al., 2009]

How shall we branch?

Integer variables

- Integrate directly into a MIP solver?
  - SCIP [Vigerske and Gleixner, 2018] • CPLEX [Bonami et al., 2016]
- Leave the discrete variables to a state-of-the-art MIP solver?
  - BONMIN [Bonami et al., 2008] • ANTIGONE [Misener and Floudas, 2014]
- Effectively develop a MIP solver together with the MINLP solver?
  - BARON [Tawarmalani and Sahinidis, 2005] • Couenne [Belotti et al., 2009]

Continuous variables

- On the original problem variables?
- In a reduced space?
  - Concave [Phillips and Rosen, 1988] • Bilinear [Sherali and Alameddine, 1992] • Difference of convex [Horst and Van Thoai, 1994] • Factorable MINLP [Epperly and Pistikopoulos, 1997] • Flowsheets [Bongartz and Mitsos, 2017]
- In a lifted space?

How shall we branch?

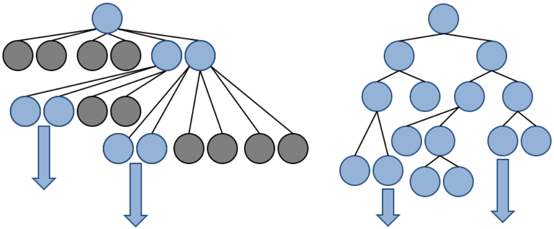
$$\begin{aligned} \min_{\mathbf{x}_1,\mathbf{x}_2} \quad & f^{\text{cvx}}(\mathbf{x}_1,\mathbf{x}_2) \\ & g_1^{\text{cvx}}(\mathbf{x}_1,\mathbf{x}_2) + \textstyle\sum_i g_{2,i}^{\text{cvx}}(\mathbf{x}_1)\cdot g_{3,i}^{\text{noncvx}}(\mathbf{x}_2) + g_4^{\text{noncvx}}(\mathbf{x}_2) \le 0 \\ & \mathbf{x}_1 \in \left[\mathbf{x}_1^{\mathrm{L}},\, \mathbf{x}_1^{\mathrm{U}}\right] \\ & \mathbf{x}_2 \in \left[\mathbf{x}_2^{\mathrm{L}},\, \mathbf{x}_2^{\mathrm{U}}\right] \end{aligned}$$

Only need to branch on the $\mathbf{x}_2$ variables!

Continuous variables

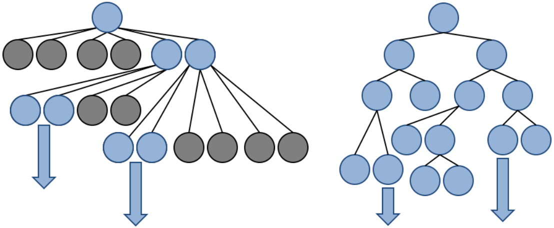
- On the original problem variables?
- In a reduced space?
  - Concave [Phillips and Rosen, 1988] • Bilinear [Sherali and Alameddine, 1992] • Difference of convex [Horst and Van Thoai, 1994] • **Factorable MINLP** [Epperly and Pistikopoulos, 1997] • Flowsheets [Bongartz and Mitsos, 2017]
- In a lifted space?

Reliability branching[28]

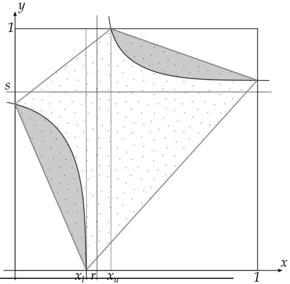


[28] Achterberg et al. [2005], Belotti et al. [2009], Misener and Floudas [2014], Vigerske and Gleixner [2018]

Reliability branching[28]



Get disjunction [Dey et al., 2019]



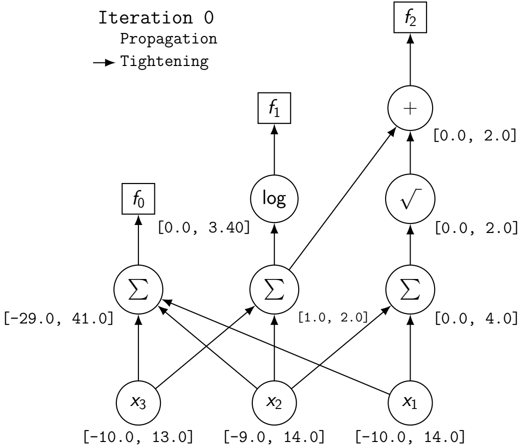
Outlook

Get disjunctions • Branch on constraints • Improve relaxations & bounds tightening

[28] Achterberg et al. [2005], Belotti et al. [2009], Misener and Floudas [2014], Vigerske and Gleixner [2018]

Outline

Feasibility & optimality-based bounds tightening[29]

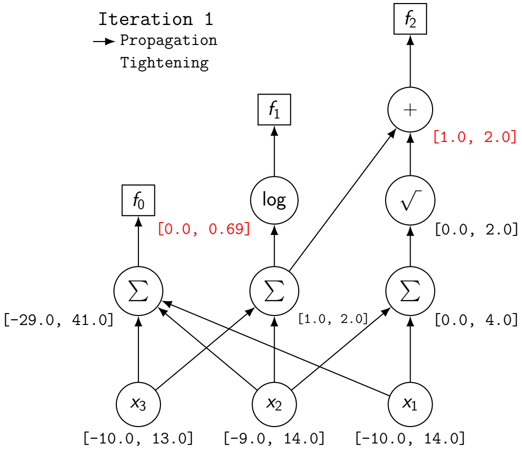


Outlook

Major accomplishments enabled generic solver software • New challenge: tighten more than variables

[29] Ryoo and Sahinidis [1996], Zamora and Grossmann [1999], Belotti et al. [2009], Belotti [2013], Gleixner et al. [2017], Puranik and Sahinidis [2017a,b]

Feasibility & optimality-based bounds tightening[29]

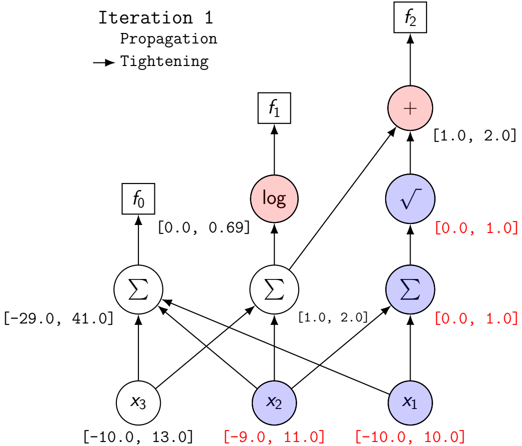


Outlook

Major accomplishments enabled generic solver software • New challenge: tighten more than variables

[29] Ryoo and Sahinidis [1996], Zamora and Grossmann [1999], Belotti et al. [2009], Belotti [2013], Gleixner et al. [2017], Puranik and Sahinidis [2017a,b]

Feasibility & optimality-based bounds tightening[29]

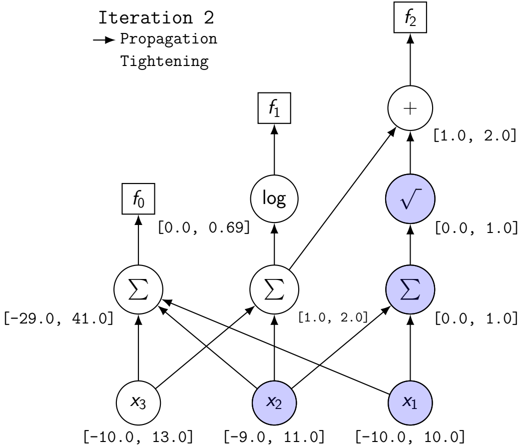


Outlook

Major accomplishments enabled generic solver software • New challenge: tighten more than variables

[29] Ryoo and Sahinidis [1996], Zamora and Grossmann [1999], Belotti et al. [2009], Belotti [2013], Gleixner et al. [2017], Puranik and Sahinidis [2017a,b]

Feasibility & optimality-based bounds tightening[29]



Outlook

Major accomplishments enabled generic solver software • New challenge: tighten more than variables

[29] Ryoo and Sahinidis [1996], Zamora and Grossmann [1999], Belotti et al. [2009], Belotti [2013], Gleixner et al. [2017], Puranik and Sahinidis [2017a,b]

Outline

Primal Heuristics: Overview

Ask the Jedi Masters . . .

- Fischetti and Lodi [2010]
- Berthold [2014] says to *fail fast*

Machine learning for heuristics [Bengio, Lodi, and Prouvost, 2018]

- **Offline learning** to decide a selection of heuristics [Khalil et al., 2017]
- **Online learning** to chose a heuristic [Hendel, 2018] or guide local search [Shylo and Shams, 2018]
- **Specific problem classes** [Bello et al., 2016, Hottung et al., 2020, Hu et al., 2018, Fischetti and Fraccaro, 2019, Kool et al., 2019]

Outline

RLT & SDP relaxations

Anstreicher [2009]

$$\begin{aligned} \min_x \quad & x^T Q x + c^T x \\ & Ax \le b, \\ \\ & x \in [0,1]^N \end{aligned}$$

- $X_{ij} = X_{ji}$,
- $Q \bullet X$ is the matrix inner product $Q \bullet X = \sum_{i,j=1}^{N} Q_{ij} \cdot X_{ij}$,
- RLT $\equiv$ Reformulation linearisation technique.
- SDP $\equiv$ Semidefinite programming,

RLT & SDP relaxations

Anstreicher [2009]

$$\begin{aligned} \min_x \quad & Q \bullet X + c^T x \\ & Ax \leq b, \\ & X = xx^T \\ & x \in [0,1]^N \\ & X \in [0,1]^{N \times N} \end{aligned}$$

- $X_{ij} = X_{ji}$,
- $Q \bullet X$ is the matrix inner product $Q \bullet X = \sum_{i,j=1}^{N} Q_{ij} \cdot X_{ij}$,
- RLT $\equiv$ Reformulation linearisation technique.
- SDP $\equiv$ Semidefinite programming,

RLT & SDP relaxations

Anstreicher [2009]

$$
\begin{aligned}
\min_x \quad & Q \bullet X + c^T x \\
& Ax \le b, \\
& X = xx^T \\
& x \in [0,1]^N \\
& X \in [0,1]^{N\times N}
\end{aligned}
\quad \Rightarrow \quad
\begin{aligned}
\min_x \quad & Q \bullet X + c^T x \\
& Ax \le b, \\
& \mathbf{X_{ij} - x_i - x_j \ge -1} \\
& \mathbf{X_{ij} - x_i \le 0} && \textbf{RLT relaxation} \\
& \mathbf{X_{ij} - x_j \le 0} \\
& X \succeq xx^T && \text{SDP relaxation} \\
& x \in [0,1]^N \\
& X \in [0,1]^{N\times N}
\end{aligned}
$$

- $X_{ij} = X_{ji}$,
- $Q \bullet X$ is the matrix inner product $Q \bullet X = \sum_{i,j=1}^{N} Q_{ij} \cdot X_{ij}$,
- RLT $\equiv$ Reformulation linearisation technique.
- SDP $\equiv$ Semidefinite programming,

RLT & SDP relaxations

Anstreicher [2009]

$$
\begin{array}{llcll}
\min\limits_x & Q \bullet X + c^T x & \Rightarrow & \min\limits_x & Q \bullet X + c^T x \\
& Ax \leq b, & & & Ax \leq b, \\
& X = xx^T & & & \mathbf{X_{ij} - x_i - x_j \geq -1} \\
& x \in [0,1]^N & & & \mathbf{X_{ij} - x_i \leq 0} \quad \textbf{RLT relaxation} \\
& X \in [0,1]^{N \times N} & & & \mathbf{X_{ij} - x_j \leq 0} \\
& & & & \mathbf{X \succeq xx^T} \quad \textbf{SDP relaxation} \\
& & & & x \in [0,1]^N \\
& & & & X \in [0,1]^{N \times N}
\end{array}
$$

- $X_{ij} = X_{ji}$,
- $Q \bullet X$ is the matrix inner product $Q \bullet X = \sum_{i,j=1}^{N} Q_{ij} \cdot X_{ij}$,
- RLT $\equiv$ Reformulation linearisation technique.
- SDP $\equiv$ Semidefinite programming,

RLT & SDP to RLT & PSD

Sherali and Fraticelli [2002], Qualizza et al. [2012]

$$\begin{aligned}
\min_x \quad & Q \bullet X + c^T x \\
& Ax \leq b, \\
& \mathbf{X_{ij} - x_i - x_j \geq -1} \\
& \mathbf{X_{ij} - x_i \leq 0} \\
& \mathbf{X_{ij} - x_j \leq 0} \\
& \mathbf{X \succeq xx^T} \\
& \\
& x \in [0,1]^N \\
& X \in [0,1]^{N \times N}
\end{aligned}$$

- RLT $\equiv$ Reformulation linearization technique.
- SDP $\equiv$ Semidefinite programming,

RLT & SDP to RLT & PSD

Sherali and Fraticelli [2002], Qualizza et al. [2012]

$$\begin{aligned}\min_x \quad & Q \bullet X + c^T x \\ & Ax \le b, \\ & \mathbf{X_{ij} - x_i - x_j \ge -1} \\ & \mathbf{X_{ij} - x_i \le 0} \\ & \mathbf{X_{ij} - x_j \le 0} \\ & \mathbf{X \succeq xx^T} \\ & x \in [0,1]^N \\ & X \in [0,1]^{N \times N}\end{aligned} \quad \Rightarrow \quad \begin{aligned}\min_x \quad & Q \bullet X + c^T x \\ & Ax \le b, \\ & \mathbf{X_{ij} - x_i - x_j \ge -1} \\ & \mathbf{X_{ij} - x_i \le 0} \\ & \mathbf{X_{ij} - x_j \le 0} \\ & \mathbf{v^T} \begin{bmatrix} 1 & x^T \\ x & X \end{bmatrix} \mathbf{v \ge 0} \quad \forall \mathbf{v} \in \mathbb{R}^{\mathbf{N+1}} \\ & x \in [0,1]^N \\ & X \in [0,1]^{N \times N}\end{aligned}$$

- RLT $\equiv$ Reformulation linearization technique.
- SDP $\equiv$ Semidefinite programming,

Cutting plane method for RLT & PSD

Sherali and Fraticelli [2002], Qualizza et al. [2012]

$$\min_x \quad Q \bullet X + c^T x$$
$$Ax \leq b,$$
$$\mathbf{X_{ij} - x_i - x_j \geq -1}$$
$$\mathbf{X_{ij} - x_i \leq 0}$$
$$\mathbf{X_{ij} - x_j \leq 0}$$
$$\mathbf{v} \begin{bmatrix} 1 & x^T \\ x & X \end{bmatrix} \mathbf{v^T} \geq \mathbf{0} \quad \forall \mathbf{v} \in \mathbb{R}^{\mathbf{N+1}}$$
$$x \in [0,1]^N$$
$$X \in [0,1]^{N \times N}$$

- **Solve RLT only**, get $\tilde{X}, \tilde{x}$
- Add other polyhedral cuts, e.g. **triangle inequalities, edge-concave cuts,** $\{0, 1/2\}$**-cuts**
- Eigendecomposition on $\begin{bmatrix} 1 & \tilde{x}^T \\ \tilde{x} & \tilde{X} \end{bmatrix}$ yields t negative eigenvalues.
  - If $t = 0$, terminate.
  - If $t \geq 1$, add cuts $\mathbf{v_k} \begin{bmatrix} 1 & x^T \\ x & X \end{bmatrix} \mathbf{v_k^T} \geq \mathbf{0} \ \forall \mathbf{k} = \mathbf{1}, \ldots, \mathbf{t}$

Other cuts to integrate ...

Frangioni and Gentile [2006], Bao et al. [2009], Misener et al. [2015], Bonami et al. [2016], Baltean-Lugojan and Misener [2018], Del Pia et al. [2018]

Outline

Challenges [with some ideas] . . .

- **Data structures**
  - Move beyond boxes [Kvasnica et al., 2004] • Trade-off between recognition & disciplined programming? [Ceccon et al., 2020]
- **Convex relaxations**
  - Move beyond boxes [Slide 34] • How many auxiliary variables to introduce? • Manage symmetry [Müller et al., 2020]
- **Branching**
  - Get disjunctions [Dey et al., 2019] • Managing integers • Branch on constraints • Learning [Lodi and Zarpellon, 2017, Dilkina et al., 2017]
- **Bounds tightening**
  - Incorporate multiple constraints [Belotti, 2013] • Tighter integration with convex relaxations or branching?
- **Integrating machine learning** [Bengio, Lodi, and Prouvost, 2018]

Challenges [with some ideas] . . .

- **Data structures**
  - Move beyond boxes [Kvasnica et al., 2004] • Trade-off between recognition & disciplined programming? [Ceccon et al., 2020]
- **Convex relaxations**
  - Move beyond boxes [Slide 34] • How many auxiliary variables to introduce? • Manage symmetry [Müller et al., 2020]
- **Branching**
  - Get disjunctions [Dey et al., 2019] • Managing integers • Branch on constraints • Learning [Lodi and Zarpellon, 2017, Dilkina et al., 2017]
- **Bounds tightening**
  - Incorporate multiple constraints [Belotti, 2013] • Tighter integration with convex relaxations or branching?
- **Integrating machine learning** [Bengio, Lodi, and Prouvost, 2018]

My questions . . .

What is missing on this list? • Anything here already easy to manage?

References I

K. Abhishek, S. Leyffer, and J. Linderoth. Filmint: An outer approximation-based solver for convex mixed-integer nonlinear programs. *INFORMS J Comput*, 22 (4):555–567, 2010.

T Achterberg and C Raack. The MCF-separator: Detecting and exploiting multi-commodity flow structures in MIPs. *Math Program Comput*, 2(2):125–165, 2010.

T. Achterberg, T. Koch, and A. Martin. Branching rules revisited. *Oper Res Letters*, 33(1):42–54, 2005.

W. P. Adams and H. D. Sherali. Linearization strategies for a class of zero-one mixed integer programming problems. *Oper Res*, 38(2):217–226, 1990.

N. Adhya, M. Tawarmalani, and N. V. Sahinidis. A Lagrangian approach to the pooling problem. *Ind Eng Chem Res*, 38(5):1956–1972, 1999.

C. S. Adjiman, I. P. Androulakis, and C. A. Floudas. A global optimization method, αBB, for general twice differentiable NLPs-II. Implementation and computational results. *Comput Chem Eng*, 22:1159–1179, 1998a.

C. S. Adjiman, S. Dallwig, C. A. Floudas, and A. Neumaier. A global optimization method, αBB, for general twice-differentiable constrained NLPsi. theoretical advances. *Comput Chem Eng*, 22(9):1137–1158, 1998b.

A. A. Ahmadi and A. Majumdar. DSOS and SDSOS optimization: more tractable alternatives to sum of squares and semidefinite optimization. *SIAM J Applied Algebra and Geometry*, 3(2):193–230, 2019.

M. S. Aktürk, A. Atamtürk, and S. Gürel. A strong conic quadratic reformulation for machine-job assignment with controllable processing times. *Oper Res Letters*, 37(3):187–191, 2009.

F. A. Al-Khayyal and J. E. Falk. Jointly constrained biconvex programming. *Math Oper Res*, 8(2):273–286, 1983.

M. Alfaki and D. Haugland. Strong formulations for the pooling problem. *J Glob Optim*, 56(3):897–916, 2013.

I. P. Androulakis, C. D. Maranas, and C. A. Floudas. αBB: A global optimization method for general constrained nonconvex problems. *J Glob Optim*, 7(4): 337–363, 1995.

K. M. Anstreicher. Semidefinite programming versus the reformulation-linearization technique for nonconvex quadratically constrained quadratic programming. *J Glob Optim*, 43(2):471–484, 2009.

K. M. Anstreicher and S. Burer. Computable representations for convex hulls of low-dimensional quadratic forms. *Math Program*, 124(1-2):33–43, 2010.

K. M. Anstreicher, S. Burer, and K. Park. Convex hull representations for bounded products of variables. *arXiv preprint arXiv:2004.07233*, 2020.

M. Ballerstein and D. Michaels. Extended formulations for convex envelopes. *J Glob Optim*, 60(2):217–238, 2014.

References II

R. Baltean-Lugojan and R. Misener. Piecewise parametric structure in the pooling problem: from sparse strongly-polynomial solutions to NP-hardness. *J Glob Optim*, 71(4):655–690, 2018.

R. Baltean-Lugojan, P. Bonami, R. Misener, and A. Tramontani. Selecting cutting planes for quadratic semidefinite outer-approximation via trained neural networks. 2018.

X. Bao, N. V. Sahinidis, and M. Tawarmalani. Multiterm polyhedral relaxations for nonconvex, quadratically constrained quadratic programs. *Optim Method Softw*, 24(4-5):485–504, 2009.

X. Bao, A. Khajavirad, N. V. Sahinidis, and M. Tawarmalani. Global optimization of nonconvex problems with multilinear intermediates. *Math Program Comput*, 7(1):1–37, 2015.

A. Bärmann, T. Gellermann, M. Merkert, and O. Schneider. Staircase compatibility and its applications in scheduling and piecewise linearization. *Discrete Optimization*, 29:111 – 132, 2018.

P. I. Barton and C. C. Pantelides. Modeling of combined discrete/continuous processes. *AIChE J*, 40(6):966–979, 1994.

I. Bello, H. Pham, Q. V. Le, M. Norouzi, and S. Bengio. Neural combinatorial optimization with reinforcement learning. *arXiv preprint arXiv:1611.09940*, 2016.

P. Belotti. Bound reduction using pairs of linear inequalities. *J Glob Optim*, 56(3):787–819, 2013.

P. Belotti, J. Lee, L. Liberti, F. Margot, and A. Wächter. Branching and bounds tightening techniques for non-convex MINLP. *Optim Method Softw*, 24(4-5): 597–634, 2009.

P. Belotti, C. Kirches, S. Leyffer, J. Linderoth, J. Luedtke, and A. Mahajan. Mixed-integer nonlinear optimization. *Acta Numerica*, 22:1–131, 2013.

Y. Bengio, A. Lodi, and A. Prouvost. Machine learning for combinatorial optimization: a methodological tour d'horizon. *arXiv preprint arXiv:1811.06128*, 2018.

H. Y. Benson. Mixed integer nonlinear programming using interior-point methods. *Optim Method Softw*, 26(6):911–931, 2011.

T. Berthold. *Heuristic algorithms in global MINLP solvers*. Verlag Dr. Hut Munich, 2014.

D. Bienstock and G. Munoz. On linear relaxations of OPF problems. *arXiv preprint arXiv:1411.1120*, 2014.

R. E. Bixby and R. Fourer. Finding embedded network rows in linear programs I. extraction heuristics. *Manag Sci*, 34(3):342–376, 1988.

N. Boland, T. Kalinowski, and F. Rigterink. New multi-commodity flow formulations for the pooling problem. *J Glob Optim*, 66(4):669–710, 2016.

References III

G. M. Bollas, P. I. Barton, and A. Mitsos. Bilevel optimization formulation for parameter estimation in vaporliquid(liquid) phase equilibrium problems. *Chem Eng Sci*, 64(8):1768 – 1783, 2009.

A. Bompadre and A. Mitsos. Convergence rate of McCormick relaxations. *J Glob Optim*, 52(1):1–28, 2012.

P. Bonami, L. T. Biegler, A. R. Conn, G. Cornuéjols, I. E. Grossmann, C. D. Laird, J. Lee, A. Lodi, F. Margot, N. Sawaya, and A. Wächter. An algorithmic framework for convex mixed integer nonlinear programs. *Discrete Optim*, 5(2):186–204, 2008.

P. Bonami, O. Günlük, and J. Linderoth. Solving box-constrained nonconvex quadratic programs. 2016. URL `http://www.optimization-online.org/DB_FILE/2016/06/5488.pdf`.

D. Bongartz and A. Mitsos. Deterministic global optimization of process flowsheets in a reduced space using mccormick relaxations. *J Glob Optim*, 69(4): 761–796, 2017.

F. Boukouvala, R. Misener, and C. A. Floudas. Global optimization advances in mixed-integer nonlinear programming, MINLP, and constrained derivative-free optimization, CDFO. *Eur J Oper Res*, 252(3):701–727, 2016.

A. Brooke, D. Kendrick, A. Meeraus, and R. Raman. *GAMS Language Guide*. Number 1. Citeseer, 1997.

G. G. Brown and W. G. Wright. Automatic identification of embedded network rows in large-scale optimization models. *Math Program*, 29(1):41 – 56, 1984.

S. Burer and F. Kılınç-Karzan. How to convexify the intersection of a second order cone and a nonconvex quadratic. *Math Program*, 162(1-2):393–429, 2017.

M. R. Bussieck and S. Vigerske. MINLP solver software. In J. J. Cochran, L. A. Cox, P. Keskinocak, J. P. Kharoufeh, and J. C. Smith, editors, *Wiley Encyclopedia of Operations Research and Management Science*. John Wiley & Sons, Inc., 2010.

M. R. Bussieck, A. S. Drud, and A. Meeraus. MINLPLib - a collection of test models for mixed-integer nonlinear programming. *INFORMS J Comput*, 15(1), 2003.

R. H. Byrd, J. Nocedal, and R. A. Waltz. Knitro: An integrated package for nonlinear optimization. In G. Di Pillo and M. Roma, editors, *Large-Scale Nonlinear Optimization*, pages 35–59. Springer US, Boston, MA, 2006.

S. Cafieri, J. Lee, and L. Liberti. On convex relaxations of quadrilinear terms. *J Glob Optim*, 47(4):661–685, 2010.

F. Ceccon, G. Kouyialis, and R. Misener. Using functional programming to recognize named structure in an optimization problem: Application to pooling. *AIChE J*, 62(9):3085–3095, 2016.

F. Ceccon, J. D. Siirola, and R. Misener. SUSPECT: MINLP special structure detector for Pyomo. *Optim Lett*, 14:801–814, 2020.

References IV

S. Ceria and J. Soares. Convex programming for disjunctive convex optimization. *Math Program*, 86(3):595–614, 1999.

C. Chen, A. Atamtürk, and S. S. Oren. Bound tightening for the alternating current optimal power flow problem. *IEEE Transactions on Power Systems*, 31(5): 3729–3736, 2015.

J. W. Chinneck. Analyzing Mathematical Programs Using MProbe. *Annals Oper Res*, 104(1–4):33–48, 2001.

C. Coey, M. Lubin, and J. P. Vielma. Outer approximation with conic certificates for mixed-integer convex problems. *Mathematical Programming Computation*, pages 1–45, 2020.

C. D'Ambrosio and A. Lodi. Mixed integer nonlinear programming tools: a practical overview. *4OR*, 9(4):329–349, 2011.

D. Davarnia, J.-P. P. Richard, and M. Tawarmalani. Simultaneous convexification of bilinear functions over polytopes with application to network interdiction. *SIAM J Optim*, 27(3):1801–1833, 2017.

A. Del Pia and A. Khajavirad. The running intersection relaxation of the multilinear polytope. *Optimization Online*, 6618, 2018.

A. Del Pia, A. Khajavirad, and N. V. Sahinidis. On the impact of running intersection inequalities for globally solving polynomial optimization problems. *Math Program Comput*, 12:165–191, 2018.

S. S. Dey and A. Gupte. Analysis of MILP techniques for the pooling problem. *Oper Res*, 63(2):412–427, 2015.

S. S. Dey, A. Santana, and Y. Wang. New SOCP relaxation and branching rule for bipartite bilinear programs. *Optim Eng*, 20(2):307–336, 2019.

S. S. Dey, B. Kocuk, and A. Santana. Convexifications of rank-one-based substructures in QCQPs and applications to the pooling problem. *J Glob Optim*, 77: 227–272, 2020.

B. Dilkina, E. B. Khalil, and G. L. Nemhauser. Comments on: On learning and branching: a survey. *TOP*, 25(2):242–246, 2017.

I. Dunning, J. Huchette, and M. Lubin. JuMP: A Modeling Language for Mathematical Optimization. *SIAM Review*, 59(2):295–320, 2017.

A. Eltved, J. Dahl, and M. S. Andersen. On the robustness and scalability of semidefinite relaxation for optimal power flow problems. *Optim Eng*, 2019.

T. G. W. Epperly and E. N. Pistikopoulos. A reduced space branch and bound algorithm for global optimization. *J Glob Optim*, 11(3):287–311, 1997.

M. Escobar and I. Grossmann. Mixed-integer nonlinear programming models for optimal simultaneous synthesis of heat exchangers network, 2010. Available from CyberInfrastructure for MINLP [`www.minlp.org`, a collaboration of Carnegie Mellon University and IBM Research] at: `www.minlp.org/library/problem/index.php?i=93`.

References V

L.B. Evans, J.F. Boston, H.I. Britt, P.W. Gallier, P.K. Gupta, B. Joseph, V. Mahalec, E. Ng, W.D. Seider, and H. Yagi. ASPEN: An Advanced System for Process Engineering. *Comput Chem Eng*, 3(1):319 – 327, 1979.

M. Fischetti and M. Fraccaro. Machine learning meets mathematical optimization to predict the optimal production of offshore wind parks. *Comput Oper Res*, 106:289–297, 2019.

M. Fischetti and A. Lodi. Heuristics in mixed integer programming. *Wiley Encyclopedia of Operations Research and Management Science*, 2010.

Roger Fletcher and Sven Leyffer. Numerical experience with lower bounds for MIQP branch-and-bound. *SIAM J Optim*, 8(2):604–616, 1998.

C. A. Floudas and C. E. Gounaris. A review of recent advances in global optimization. *J Glob Optim*, 45(1):3 – 38, 2009.

R. Fourer, D. Gay, and B. Kernighan. *AMPL*. Boyd & Fraser Danvers, MA, 1993.

R. Fourer, C. Maheshwari, A. Neumaier, D. Orban, and H. Schichl. Convexity and concavity detection in computational graphs: Tree walks for convexity assessment. *INFORMS J Comput*, 22(1):26–43, 2010.

A. Frangioni and C. Gentile. Perspective cuts for a class of convex 0–1 mixed integer programs. *Math Program*, 106(2):225–236, 2006.

M. Fukuda, M. Kojima, K. Murota, and K. Nakata. Exploiting sparsity in semidefinite programming via matrix completion I: General framework. *SIAM J Optim*, 11(3):647–674, 2001.

C. Y. Gau and L. E. Schrage. Implementation and testing of a branch-and-bound based method for deterministic global optimization: Operations research applications. In *Frontiers in global optimization*, pages 145–164. Springer, Boston, MA, 2004.

M. Glass and A. Mitsos. Thermodynamic analysis of formulations to discriminate multiple roots of cubic equations of state in process models. *Comput Chem Eng*, 106:407 – 420, 2017. ESCAPE-26.

A. M. Gleixner, T. Berthold, B. Müller, and S. Weltge. Three enhancements for optimization-based bound tightening. *J Glob Optim*, 67(4):731–757, 2017.

M. Grant and S. Boyd. Graph implementations for nonsmooth convex programs. In V. Blondel, S. Boyd, and H. Kimura, editors, *Recent Advances in Learning and Control*, Lecture Notes in Control and Information Sciences, pages 95–110. Springer-Verlag Limited, 2008. `http://stanford.edu/~boyd/graph_dcp.html`.

I. E. Grossmann and L. T. Biegler. Part II. Future perspective on optimization. *Comput. Chem. Eng.*, 28(8):1193 – 1218, 2004.

I. E. Grossmann and R. W. H. Sargent. Optimum design of multipurpose chemical plants. *Industrial & Engineering Chemistry Process Design and Development*, 18(2):343–348, 1979.

References VI

G. Guillén-Gosálbez and I. Grossmann. A global optimization strategy for the environmentally conscious design of chemical supply chains under uncertainty in the damage assessment model. *Comput Chem Eng*, 34(1):42 – 58, 2010.

A. Gupte, S. Ahmed, S. S. Dey, and M. S. Cheon. Relaxations and discretizations for the pooling problem. *J Glob Optim*, 67(3):631–669, 2017.

W. E. Hart, C. D. Laird, J.-P. Watson, D. L. Woodruff, G. A. Hackebeil, B. L. Nicholson, and J. D. Siirola. *Pyomo–optimization modeling in python*. Springer Science & Business Media, second edition, 2017.

M. M. F. Hasan, R. C. Baliban, J. A. Elia, and C. A. Floudas. Modeling, simulation, and optimization of postcombustion CO2 capture for variable feed concentration and flow rate. 2. pressure swing adsorption and vacuum swing adsorption processes. *Ind Eng Chem Res*, 51(48):15665–15682, 2012.

G. Hendel. Adaptive large neighborhood search for mixed integer programming. Technical Report 18-60, ZIB, Takustr. 7, 14195 Berlin, 2018.

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

H. Hijazi, P. Bonami, and A. Ouorou. An outer-inner approximation for separable mixed-integer nonlinear programs. *INFORMS J Comput*, 26(1):31–44, 2013.

R. Horst and N. Van Thoai. Constraint decomposition algorithms in global optimization. *J Glob Optim*, 5(4):333–348, 1994.

A. Hottung, S. Tanaka, and K. Tierney. Deep learning assisted heuristic tree search for the container pre-marshalling problem. *Comput Oper Res*, 113:104781, 2020.

H. Hu, L. Duan, X. Zhang, Y. Xu, and J. Wei. A multi-task selected learning approach for solving new type 3D bin packing problem. *CoRR*, abs/1804.06896, 2018.

J. Humpola and A. Fügenschuh. Convex reformulations for solving a nonlinear network design problem. *Comput Optim Appl*, 62(3):717–759, 2015.

M. Hunting. The AIMMS outer approximation algorithm for MINLP. *Paragon Decision Technology, Haarlem*, 2011.

D. R. Jones, C. D. Perttunen, and B. E. Stuckman. Lipschitzian optimization without the Lipschitz constant. *J Optim Theory Appl*, 79(1):157–181, 1993.

R. Karuppiah and I. E. Grossmann. Global optimization for the synthesis of integrated water systems in chemical processes. *Comput Chem Eng*, 30(4):650–673, 2006.

R Baker Kearfott. *Rigorous global search: continuous problems*. 1996.

A. Khajavirad and N. V. Sahinidis. Convex envelopes generated from finitely many compact convex sets. *Math Program*, 137(1):371–408, 2013.

E. B. Khalil, B. Dilkina, G. L. Nemhauser, S. Ahmed, and Y. Shao. Learning to run heuristics in tree search. In *IJCAI*, pages 659–666, 2017.

References VII

T. Koch. *Rapid Mathematical Prototyping*. PhD thesis, Technische Universität Berlin, 2004.

B. Kocuk, S. S. Dey, and X. A. Sun. Strong SOCP relaxations for the optimal power flow problem. *Oper Res*, 64(6):1177–1196, 2016.

W. Kool, H. van Hoof, and M. Welling. Attention, learn to solve routing problems! In *ICLR 2019*, 2019.

O. Kröger, C. Coffrin, H. Hijazi, and H. Nagarajan. Juniper: An open-source nonlinear branch-and-bound solver in Julia. In W.-J. van Hoeve, editor, *CPAIOR*, pages 377–386. Springer International Publishing, 2018.

J. Kronqvist, D. E. Bernal, A. Lundell, and I. E. Grossmann. A review and comparison of solvers for convex MINLP. *Optim Eng*, 20(2):397–455, 2019.

M. Kvasnica, P. Grieder, M. Baotić, and M. Morari. Multi-parametric toolbox (mpt). In *International Workshop on Hybrid Systems: Computation and Control*, pages 448–462, 2004.

J. B. Lasserre. Global optimization with polynomials and the problem of moments. *SIAM J Optim*, 11(3):796–817, 2001.

S. Le Digabel. Algorithm 909: NOMAD: Nonlinear optimization with the MADS algorithm. *ACM Transactions on Mathematical Software (TOMS)*, 37(4):44, 2011.

S. Lee and I. E. Grossmann. New algorithms for nonlinear generalized disjunctive programming. *Comput Chem Eng*, 24(9-10):2125–2141, 2000.

K. T. Leperi, R. Q. Snurr, and F. You. Optimization of two-stage pressure/vacuum swing adsorption with variable dehydration level for postcombustion carbon capture. *Ind Eng Chem Res*, 55(12):3338–3350, 2016.

Q. Li and V. Vittal. Convex hull of the quadratic branch AC power flow equations and its application in radial distribution networks. *IEEE Transactions on Power Systems*, 33(1):839–850, 2018.

L. Liberti. Reformulations in mathematical programming: automatic symmetry detection and exploitation. *Math Program*, 131(1):273–304, 2012.

L. Liberti and C. C. Pantelides. Convex envelopes of monomials of odd degree. *J Glob Optim*, 25(2):157–168, 2003.

L. Liberti and C. C. Pantelides. An exact reformulation algorithm for large nonconvex nlps involving bilinear terms. *J Glob Optim*, 36(2):161–189, 2006.

L. Liberti, S. Cafieri, and D. Savourey. The reformulation-optimization software engine. In *International Congress on Mathematical Software*, pages 303–314, 2010.

F. Liers and M. Merkert. Structural investigation of piecewise linearized network flow problems. *SIAM J Optim*, 26(4):2863–2886, 2016.

J. Linderoth. A simplicial branch-and-bound algorithm for solving quadratically constrained quadratic programs. *Math Program*, 103(2):251–282, 2005.

References VIII

M. Locatelli. Convex envelopes of bivariate functions through the solution of KKT systems. *J Glob Optim*, 72(2):277–303, 2018.

A. Lodi and G. Zarpellon. On learning and branching: a survey. *TOP*, 25(2):207–236, 2017.

J. Löfberg. YALMIP: A toolbox for modeling and optimization in MATLAB. In *IEEE International Conference on Robotics and Automation (IEEE Cat. No.04CH37508)*, pages 284–289, 2004.

J. Luedtke, C. D'Ambrosio, J. Linderoth, and J. Schweiger. Strong convex nonlinear relaxations of the pooling problem. *SIAM J Optim*, 30(2):1582–1609, 2020.

A. Lundell, J. Kronqvist, and T. Westerlund. The supporting hyperplane optimization toolkit. *URL http://www.github.com/coin-or/shot*, 2018.

A. Mahajan, S. Leyffer, J. Linderoth, J. Luedtke, and T. Munson. Minotaur: A mixed-integer nonlinear optimization toolkit. 2017. URL `http://www.optimization-online.org/DB_FILE/2017/10/6275.pdf`.

C. D. Maranas and C. A. Floudas. Global optimization in generalized geometric programming. *Comput Chem Eng*, 21(4):351–369, 1997.

A. Marandi, J. Dahl, and E. de Klerk. A numerical evaluation of the bounded degree sum-of-squares hierarchy of Lasserre, Toh, and Yang on the pooling problem. *Annals of Operations Research*, pages 1–26, 2017.

G. P. McCormick. Computability of global solutions to factorable nonconvex programs: Part iconvex underestimating problems. *Math Program*, 10(1):147–175, 1976.

W. Melo, M. Fampa, and F. Raupp. An overview of MINLP algorithms and their implementation in Muriqui optimizer. *Annals of Operations Research*, 2018. doi: 10.1007/s10479-018-2872-5.

N. Mertens, C. Kunde, A. Kienle, and D. Michaels. Monotonic reformulation and bound tightening for global optimization of ideal multi-component distillation columns. *Optim Eng*, 19(2):479–514, 2018.

C. A. Meyer and C. A. Floudas. Trilinear monomials with positive or negative domains: Facets of the convex and concave envelopes. In C. A. Floudas and P. M. Pardalos, editors, *Frontiers in Global Optimization*, pages 327–352. Kluwer Academic Publishers, 2003.

C. A. Meyer and C. A. Floudas. Trilinear monomials with mixed sign domains: Facets of the convex and concave envelopes. *J Glob Optim*, 29(2):125–155, 2004.

C. A. Meyer and C. A. Floudas. Global optimization of a combinatorially complex generalized pooling problem. *AIChE J*, 52(3):1027–1037, 2006.

R. Misener and C. A. Floudas. Global optimization of mixed-integer quadratically-constrained quadratic programs (miqcqp) through piecewise-linear and edge-concave relaxations. *Math Program*, 136(1):155–182, 2012.

References IX

R. Misener and C. A. Floudas. GloMIQO: Global mixed-integer quadratic optimizer. *J Glob Optim*, 57(1):3–50, 2013.

R. Misener and C. A. Floudas. ANTIGONE: Algorithms for coNTinuous / Integer Global Optimization of Nonlinear Equations. *J Glob Optim*, 59(2-3):503–526, 2014.

R. Misener, J. P. Thompson, and C. A. Floudas. APOGEE: Global optimization of standard, generalized, and extended pooling problems via linear and logarithmic partitioning schemes. *Comput Chem Eng*, 35(5):876–892, 2011.

R. Misener, J. B. Smadbeck, and C. A. Floudas. Dynamically generated cutting planes for mixed-integer quadratically constrained quadratic programs and their incorporation into GloMIQO 2. *Optim Method Softw*, 30(1):215–249, 2015.

M. Mistry and R. Misener. Optimising heat exchanger network synthesis using convexity properties of the logarithmic mean temperature difference. *Comput Chem Eng*, 94:1 – 17, 2016.

A. Mitsos, G. M. Bollas, and P. I. Barton. Bilevel optimization formulation for parameter estimation in liquidliquid phase equilibrium problems. *Chem Eng Sci*, 64 (3):548 – 559, 2009a.

A. Mitsos, B. Chachuat, and P. I. Barton. McCormick-based relaxations of algorithms. *SIAM J Optim*, 20(2):573–601, 2009b.

S. Modaresi and J. P. Vielma. Convex hull of two quadratic or a conic quadratic and a quadratic inequality. *Math Program*, 164(1-2):383–409, 2017.

M. Mönnigmann. Fast calculation of spectral bounds for Hessian matrices on hyperrectangles. *SIAM J Matrix Analysis and Applications*, 32(4):1351–1366, 2011.

B. Müller, F. Serrano, and A. Gleixner. Using two-dimensional projections for stronger separation and propagation of bilinear terms. *SIAM J Optim*, 30(2): 1339–1365, 2020.

J. Najman and A. Mitsos. Tighter McCormick relaxations through subgradient propagation. *J Glob Optim*, 75(3):565–593, 2019.

K. Nakata, K. Fujisawa, M. Fukuda, M. Kojima, and K. Murota. Exploiting sparsity in semidefinite programming via matrix completion ii: Implementation and numerical results. *Math Program*, 95(2):303–327, 2003.

U. Nallasivam, V. H. Shah, A. A. Shenvi, J. Huff, M. Tawarmalani, and R. Agrawal. Global optimization of multicomponent distillation configurations: 2. enumeration based global minimization algorithm. *AIChE J*, 62(6):2071–2086, 2016.

G. L. Nemhauser, M. W. P. Savelsbergh, and G. C. Sigismondi. MINTO, a Mixed INTeger Optimizer. *Oper Res Lett*, 15(1):47 – 58, 1994.

References X

I. P. Nenov, D. H. Fylstra, and L. Kolev. Convexity determination in the Microsoft Excel Solver using automatic differentiation techniques. In *Fourth International Workshop on Automatic Differentiation*, 2004.

A. Neumaier. Complete search in continuous global optimization and constraint satisfaction. *Acta numerica*, 13(1):271–369, 2004.

T. T. Nguyen, M. Tawarmalani, and J.-P. P. Richard. Convexification techniques for linear complementarity constraints. In *International Conference on Integer Programming and Combinatorial Optimization*, pages 336–348, 2011.

T. T. Nguyen, J.-P. P. Richard, and M. Tawarmalani. Deriving convex hulls through lifting and projection. *Math Program*, 169(2):377–415, 2018.

I. Nowak, H. Alperin, and S. Vigerske. LaGO – an object oriented library for solving MINLPs. In C. Bliek, C. Jermann, and A. Neumaier, editors, *Global Optimization and Constraint Satisfaction*, pages 32–42, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg.

K. P. Papalexandri and E. N. Pistikopoulos. A multiperiod MINLP model for the synthesis of flexible heat and mass exchange networks. *Comput Chem Eng*, 18 (11):1125 – 1139, 1994.

F. E. Pereira, G. Jackson, A. Galindo, and C. S. Adjiman. A duality-based optimisation approach for the reliable solution of (P, T) phase equilibrium in volume-composition space. *Fluid Phase Equilibria*, 299(1):1 – 23, 2010.

M. E. Pfetsch, A. Fügenschuh, B. Geißler, N. Geißler, R. Gollmer, B. Hiller, J. Humpola, T. Koch, T. Lehmann, A. Martin, A. Morsi, J. Rövekamp, L. Schewe, M. Schmidt, R. Schultz, R. Schwarz, J. Schweiger, M. C. Stangl, C. Steinbach, S. Vigerske, and B. M. Willert. Validation of nominations in gas network optimization: models, methods, and solutions. *Optim Method Softw*, 30(1):15–53, 2015.

A. T. Phillips and J. B. Rosen. A parallel algorithm for constrained concave quadratic global minimization. *Math Program*, 42(1):421–448, 1988.

Y. Puranik and N. V. Sahinidis. Domain reduction techniques for global NLP and MINLP optimization. *Constraints*, 22(3):338–376, 2017a.

Y. Puranik and N. V. Sahinidis. Bounds tightening based on optimality conditions for nonconvex box-constrained optimization. *J Glob Optim*, 67(1):59–77, 2017b.

A. Qualizza, P. Belotti, and F. Margot. Linear programming relaxations of quadratically constrained quadratic programs. In *Mixed Integer Nonlinear Programming*, pages 407–426. 2012.

H. Rahman and A. Mahajan. Facets of a mixed-integer bilinear covering set with bounds on variables. *J Glob Optim*, 74(3):417–442, 2019.

H. S. Ryoo and N. V. Sahinidis. A branch-and-reduce approach to global optimization. *J Glob Optim*, 8(2):107–138, 1996.

N. V. Sahinidis. BARON: A general purpose global optimization software package. *J Glob Optim*, 8(2):201–205, 1996.

References XI

D. Salvagnin. Detecting semantic groups in MIP models. In *Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems*, Lecture Notes in Computer Science. Springer International Publishing, 2016.

A. Santana and S. S. Dey. The convex hull of a quadratic constraint over a polytope. *arXiv preprint arXiv:1812.10160*, 2018.

M. Schulze Darup and M. Mönnigmann. Improved automatic computation of Hessian matrix spectral bounds. *SIAM J Sci Comput*, 38(4):A2068–A2090, 2016.

H. D. Sherali and A. Alameddine. A new reformulation-linearization technique for bilinear programming problems. *J Glob Optim*, 2(4):379–410, 1992.

H. D. Sherali and B. M. P. Fraticelli. Enhancing RLT relaxations via a new class of semidefinite cuts. *J Glob Optim*, 22(1):233–261, 2002.

O. V. Shylo and H. Shams. Boosting binary optimization via binary classification: A case study of job shop scheduling. *arXiv preprint arXiv:1808.10813*, 2018.

E. M. B. Smith and C. C. Pantelides. A symbolic reformulation/spatial branch-and-bound algorithm for the global optimisation of nonconvex MINLPs. *Comput Chem Eng*, 23(4-5):457–478, 1999.

E. Speakman and J. Lee. Quantifying double McCormick. *Math Oper Res*, 42(4):1230–1253, 2017.

M. D. Stuber, J. K. Scott, and P. I. Barton. Convex and concave relaxations of implicit functions. *Optim Method Softw*, 30(3):424–460, 2015.

F. Tardella. On the existence of polyhedral convex envelopes. In *Frontiers in global optimization*, pages 563–573. 2004.

F. Tardella. Existence and sum decomposition of vertex polyhedral convex envelopes. *Optim Lett*, 2(3):363–375, 2008.

M. Tawarmalani and N. V. Sahinidis. Semidefinite relaxations of fractional programs via novel convexification techniques. *J Glob Optim*, 20(2):133–154, 2001.

M. Tawarmalani and N. V. Sahinidis. *Convexification and Global Optimization in Continuous and Mixed-Integer Nonlinear Programming: Theory, Algorithms, Software, and Applications*. Springer US, 2002.

M. Tawarmalani and N. V. Sahinidis. A polyhedral branch-and-cut approach to global optimization. *Math Program*, 103(2):225–249, 2005.

M. Tawarmalani, J.-P. P. Richard, and K. Chung. Strong valid inequalities for orthogonal disjunctions and bilinear covering sets. *Math Program*, 124(1-2): 481–512, 2010.

M. Tawarmalani, J.-P. P. Richard, and C. Xiong. Explicit convex and concave envelopes through polyhedral subdivisions. *Math Program*, 138(1):531–577, 2013.

A. Tsoukalas and A. Mitsos. Multivariate McCormick relaxations. *J Glob Optim*, 59(2-3):633–662, 2014.

References XII

J. P. Vielma, I. Dunning, J. Huchette, and M. Lubin. Extended formulations in mixed integer conic quadratic programming. *Math Program Comput*, 9(3): 369–418, 2017.

S. Vigerske and A. Gleixner. SCIP: Global optimization of mixed-integer nonlinear programs in a branch-and-cut framework. *Optim Method Softw*, 33(3):563–593, 2018.

S. Vigerske, S. Heinz, A. Gleixner, and T. Berthold. Analyzing the computational impact of MIQCP solver components. *Numerical Algebra, Control and Optimization*, 2(4):739–748, 2012.

J. Viswanathan and I.E. Grossmann. A combined penalty function and outer-approximation method for MINLP optimization. *Comput Chem Eng*, 14(7):769 – 782, 1990.

V. Visweswaran and C. A. Floudas. A global optimization algorithm (GOP) for certain classes of nonconvex NLPsII. Application of theory and test problems. *Comput Chem Eng*, 14(12):1419–1434, 1990.

T. Westerlund and R. Pörn. Solving pseudo-convex mixed integer optimization problems by cutting plane techniques. *Optim Eng*, 3(3):253–280, 2002.

M. Wilhelm and M. D. Stuber. Easy Advanced Global Optimization (EAGO): An open-source platform for robust and global optimization in julia. In *AIChE Annual Meeting*, 2017.

J. M. Zamora and I. E. Grossmann. A branch and contract algorithm for problems with concave univariate, bilinear and linear fractional terms. *J Glob Optim*, 14 (3):217–249, 1999.

K. Zorn and N. V. Sahinidis. Global optimization of general non-convex problems with intermediate bilinear substructures. *Optim Method Softw*, 29(3):442–462, 2014.