Mixed-Integer Nonlinear Optimization: Applications, Algorithms, and Computation IV

Sven Leyffer

Mathematics & Computer Science Division
Argonne National Laboratory

Graduate School in
Systems, Optimization, Control and Networks
Université catholique de Louvain
February 2013

Outline

Outline

The Chvátal-Gomory Procedure

A general procedure for generating valid inequalities for ILP

$$\underset{x}{\text{minimize}} \ c^T x \quad \text{subject to } Ax \le b, \ x \ge 0, \ x \in \mathbb{Z}^n$$

- Let the columns of $A \in \mathbb{R}^{m \times n}$ be denoted by $\{a_1, a_2, \ldots a_n\}$
- $S = \{x \in \mathbb{Z}^n_+ \mid Ax \le b\}$ feasible set of ILP.
  1. Choose nonnegative multipliers $u \in \mathbb{R}^m_+$
  2. $u^T Ax \le u^T b$ is a valid inequality: $\sum_{j \in N} u_j a_j x_j \le u^T b$.
  3. $\sum_{j \in N} \lfloor u^T a_j \rfloor x_j \le u^T b$, since $x \ge 0$.
  4. $\sum_{j \in N} \lfloor u^T a_j \rfloor x_j \le \lfloor u^T b \rfloor$ is valid for S
     since $\lfloor u^T a_j \rfloor x_j$ is an integer

The Chvátal-Gomory Procedure

A general procedure for generating valid inequalities for ILP

$$\underset{x}{\text{minimize}} \; c^T x \quad \text{subject to } Ax \le b,\; x \ge 0,\; x \in \mathbb{Z}^n$$

- Let the columns of $A \in \mathbb{R}^{m \times n}$ be denoted by $\{a_1, a_2, \ldots a_n\}$
- $S = \{x \in \mathbb{Z}^n_+ \mid Ax \le b\}$ feasible set of ILP.
  1. Choose nonnegative multipliers $u \in \mathbb{R}^m_+$
  2. $u^T Ax \le u^T b$ is a valid inequality: $\displaystyle\sum_{j \in N} u_j a_j x_j \le u^T b$.
  3. $\displaystyle\sum_{j \in N} \lfloor u^T a_j \rfloor x_j \le u^T b$, since $x \ge 0$.
  4. $\displaystyle\sum_{j \in N} \lfloor u^T a_j \rfloor x_j \le \lfloor u^T b \rfloor$ is valid for S since $\lfloor u^T a_j \rfloor x_j$ is an integer
- Simply Amazing: This simple procedure suffices to generate every valid inequality for an integer program

Extension to MINLP

[Çezik and Iyengar, 2005]

- This simple idea also extends to mixed 0-1 conic programming

$$\begin{cases} \underset{x}{\text{minimize}} & f^T x \\ \text{subject to} & Ax \succeq_{\mathcal{K}} b \\ & x_I \in \{0,1\}^p,\ 0 \le x \le U \end{cases}$$

- $\mathcal{K}$: Homogeneous, self-dual, proper, convex cone
- $x \succeq_{\mathcal{K}} x' \Leftrightarrow (x - x') \in \mathcal{K}$

Gomory On Cones

[Çezik and Iyengar, 2005]

- LP: $\mathcal{K}_l = \mathbb{R}^n_+$, i.e. $x \geq 0$ ... simplest cone
- SOCP: $\mathcal{K}_q = \{(x_0, \bar{x}) \mid x_0 \geq \|\bar{x}\|\}$ ... ice-cream cone
- SDP: $\mathcal{K}_s = \{x = \text{vec}(X) \mid X = X^T, X$ positive semi-definite$\}$

Gomory On Cones

[Çezik and Iyengar, 2005]

- LP: $\mathcal{K}_l = \mathbb{R}^n_+$, i.e. $x \geq 0$ ... simplest cone
- SOCP: $\mathcal{K}_q = \{(x_0, \bar{x}) \mid x_0 \geq \|\bar{x}\|\}$ ... ice-cream cone
- SDP: $\mathcal{K}_s = \{x = \text{vec}(X) \mid X = X^T, X$ positive semi-definite$\}$
- Dual Cone: $\mathcal{K}^* := \{u \mid u^T z \geq 0 \; \forall z \in \mathcal{K}\}$
- Extension is clear from the following equivalence:

$$Az \succeq_{\mathcal{K}} b \quad \Leftrightarrow \quad u^T A z \geq u^T b \; \forall u \succeq_{\mathcal{K}^*} 0$$

Gomory On Cones

[Çezik and Iyengar, 2005]

- LP: $\mathcal{K}_l = \mathbb{R}^n_+$, i.e. $x \geq 0$ ... simplest cone
- SOCP: $\mathcal{K}_q = \{(x_0, \bar{x}) \mid x_0 \geq \|\bar{x}\|\}$ ... ice-cream cone
- SDP: $\mathcal{K}_s = \{x = \text{vec}(X) \mid X = X^T, X \text{ positive semi-definite}\}$
- Dual Cone: $\mathcal{K}^* := \{u \mid u^T z \geq 0 \; \forall z \in \mathcal{K}\}$
- Extension is clear from the following equivalence:

$$Az \succeq_{\mathcal{K}} b \quad \Leftrightarrow \quad u^T Az \geq u^T b \; \forall u \succeq_{\mathcal{K}^*} 0$$

- Many classes of nonlinear inequalities can be represented as

$$Ax \succeq_{\mathcal{K}_q} b \text{ or } Ax \succeq_{\mathcal{K}_s} b$$

... e.g. perspective function $\mathcal{P}_c(x, y)$, see Part III.

Mixed-Integer Second-Order Cone Programs

Consider class of MISOCPs:

$$(\text{MISOCP}) \begin{cases} \underset{x}{\text{minimize}} & c^T x \\ \text{subject to} & x \in \mathcal{K} \\ & Ax = b,\ l \leq x \leq u \\ & x_i \in \mathbb{Z}\ \forall i \in I. \end{cases}$$

$x \in \mathcal{K}$ product of $k \geq 1$ cones $\mathcal{K} := \mathcal{K}_1 \times \ldots \times \mathcal{K}_k$, defined as

$$\mathcal{K}_j := \left\{ x_j = (x_{j0}, x_{j1}^T)^T \in \mathbb{R} \times \mathbb{R}^{n_j - 1} :\ ||x_{j1}||_2 \leq x_{j0} \right\}$$

where $x = (x_1^T, \ldots, x_k^T)^T$

Cannot apply convex MINLP solvers directly:

- Conic constraints not differentiable
- Conic constraints cause NLP solvers to fail
  ... or converge slowly

Outer Approximation for MISOCPs

For fixed integers, define SOCP subproblem:

$$
(\text{SOCP}(x_I^{(k)})) \begin{cases} \underset{x}{\text{minimize}} & c^T x \\ \text{subject to} & x \in \mathcal{K}, \\ & Ax = b, \; l \le x \le u \\ & x_I = x_I^{(k)}, \end{cases}
$$

and define outer approximations from subgradients of $\|x_{j1}\|_2 = x_{j0}$:

$$
\begin{aligned}
J_a(\bar{x}) &:= \{j : \; g_j(\bar{x}) = 0, \; \bar{x} \neq 0\}, && \text{active different.} \\
J_{0+}(\bar{x}, \bar{s}) &:= \{j : \; \bar{x}_j = 0, \; \bar{s}_{j0} > 0\}, && \text{strongly active} \\
J_{00}(\bar{x}, \bar{s}) &:= \{j : \; \bar{x}_j = 0, \; \bar{s}_{j0} = 0\}, && \text{weakly active}
\end{aligned}
$$

... and derive OA master problem ($g_j(\bar{x}) = \|x_{j1}\|_2 - x_{j0}$)

Outer Approximation for MISOCPs

Define

- $\mathcal{X}^k := \{\bar{x} : \text{ solved SOCP}(x_I^{(k)})\}$ visited points
- $U := \min\{c^T\bar{x} : \bar{x} \in \mathcal{X}^k\}$ upper bound

MISOCP outer approximation problem: (MIP($\mathcal{X}^k$))

$$\begin{cases} \underset{x}{\text{minimize}} & c^T x \\ \text{subject to} & c^T x \leq U \\ & Ax = b, \; l \leq x \leq u \\ & 0 \geq -\|\bar{x}_{j1}\| x_{j0} + \bar{x}_{j1}^T x_{j1}, \quad \forall j \in J_a(\bar{x}), \quad \bar{x} \in \mathcal{X}^k, \\ & 0 \geq -x_{j0} - \frac{1}{\bar{s}_{j0}} \bar{s}_{j1}^T x_{j1}, \quad \forall j \in J_{0+}(\bar{x}, \bar{s}), \quad \bar{x} \in \mathcal{X}^k, \\ & 0 \geq -x_{j0}, \quad \forall j \in J_{00}(\bar{x}, \bar{s}), \quad \bar{x} \in \mathcal{X}^k, \\ & x_i \in \mathbb{Z}, \quad \forall i \in I. \end{cases}$$

Convergence, see [Drewes and Ulbrich, 2012]

... Exercise: Is this OA approach finite?

Gomory Cuts for MISOCP

Theorem ([Drewes, 2009])

Continuous SOCP & dual satisfy Slater's CQ & $l_I \geq 0$.
$\bar{x}$ with $\bar{x}_I \notin \mathbb{Z}^p$ solution of SOCP($x_I^{(k)}$), $(\bar{s}, \bar{y})$ dual.
Then following cut is valid for MISOCP,

$$\lceil (A_I^T(\bar{y} - \Delta y)\bar{s}_I \rceil^T s_I \geq \lceil (\bar{y} - \Delta y)^T b \rceil,$$

where Δy solves

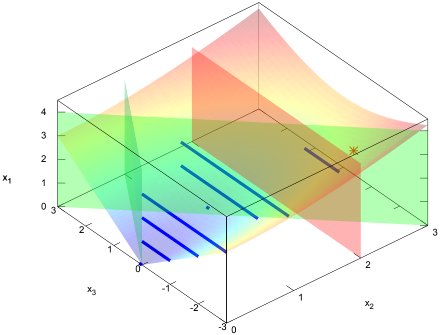
$$\begin{pmatrix} -A_C \\ A_I \end{pmatrix} \Delta y = \begin{pmatrix} c_C \\ 0 \end{pmatrix}.$$

If $(\bar{y} - \Delta y)^T b \notin \mathbb{Z}$, then cut off $\bar{x}$.

Example of Gomory for MISOCP

Example:

$$\begin{cases} \min_x -x_2 \\ \text{s.t. } -3x_2 + x_3 \leq 0 \\ \qquad 2x_2 + x_3 \leq 3 \\ \qquad 0 \leq x_1, x_2 \leq 3 \\ \qquad x_1 \geq ||(x_2, x_3)^T||_2 \\ \qquad x_1, x_2 \in \mathbb{Z}, \end{cases}$$



relaxed solution: $(3, \frac{12}{5}, -\frac{9}{5})$.
The Gomory cut $x_2 \leq 2$

Other Work on MISOCP

Related work on MISOCP (simplest generalization of MILP)

- Lift-and-project for MISOCP [Stubbs and Mehrotra, 1999] and [Drewes, 2009]
- MIR cuts for MISOCP or polyhedral SOCP [Atamtürk and Narayanan, 2010]

Outline

Challenges of Nonconvex MINLP

Mixed-Integer Nonlinear Program (MINLP)

$$\underset{x}{\text{minimize}} \; f(x) \quad \text{subject to } c(x) \leq 0, \; x \in X, \; x_i \in \mathbb{Z} \; \forall \; i \in I$$

... now drop assumption that $f(x)$ and $c(x)$ are convex

Challenges of nonconvex MINLP

- Objective function $f(x)$ can have many local minimizers
- Continuous relaxation of constraint set

$$\{x | c(x) \leq 0, \; x \in X\}$$

... can be disjoint, may have no interior

Challenges of Nonconvex MINLP

Mixed-Integer Nonlinear Program (MINLP)

$$\underset{x}{\text{minimize}} \; f(x) \quad \text{subject to } c(x) \le 0, \; x \in X, \; x_i \in \mathbb{Z} \; \forall \, i \in I$$

Nonconvexity arise naturally

- Take nonlinear, convex $c(x)$ and consider $l \le c(x) \le u$ $\Rightarrow$ nonconvex feasible region, e.g. $\{1 \le x_1^2 + x_2^2 \le 2\}$
- Nonlinear equations arise naturally in power grid applications e.g. nonlinear (AC) power flow model:

$$\begin{aligned} F(U_k, U_l, \theta_k, \theta_l) := \; & b_{kl} U_k U_l \sin(\theta_k - \theta_l) + g_{kl} U_k^2 \\ & - g_{kl} U_k U_l \cos(\theta_k - \theta_l) \end{aligned}$$

- Nonlinear equations also arise naturally in core-reloading, gas- and water-networks, and many more applications

Challenges of Nonconvex MINLP

Mixed-Integer Nonlinear Program (MINLP)

$$\underset{x}{\text{minimize}} \; f(x) \quad \text{subject to } c(x) \le 0, \; x \in X, \; x_i \in \mathbb{Z} \; \forall \, i \in I$$

Definition (Convexity)

A function $f : \mathbb{R}^n \to \mathbb{R}$ is convex, iff $\forall x^{(0)}, x^{(1)} \in \mathbb{R}^n$ we have:

$$f(x^{(1)}) \ge f(x^{(0)}) + (x^{(1)} - x^{(0)})^T \nabla f^{(0)}$$

For $f(x), c(x)$ convex we get global convergence guarantee:

- NLP relaxations ($x_i \in \mathbb{R} \; \forall \, i \in I$) are convex
  $\Rightarrow$ First-order (KKT) conditions are necessary & sufficient
  $\Rightarrow$ NLP solvers find global min at every node of BnB tree
- BnB, OA, Benders, ECP. etc. find guaranteed global solution

Challenges of Nonconvex MINLP

Mixed-Integer Nonlinear Program (MINLP)

$$\underset{x}{\text{minimize}} \ \ f(x) \quad \text{subject to } c(x) \le 0, \ x \in X, \ x_i \in \mathbb{Z} \ \forall \ i \in I$$

Definition (Convexity)

A function $f : \mathbb{R}^n \to \mathbb{R}$ is convex, iff $\forall x^{(0)}, x^{(1)} \in \mathbb{R}^n$ we have:

$$f(x^{(1)}) \ge f(x^{(0)}) + (x^{(1)} - x^{(0)})^T \nabla f^{(0)}$$

For $f(x), c(x)$ nonconvex, NLP works without guarantees:

- NLP solvers find stationary points
  $\Rightarrow$ no distinction between local/global minimum
- solution from NLP may not even be a local minimum

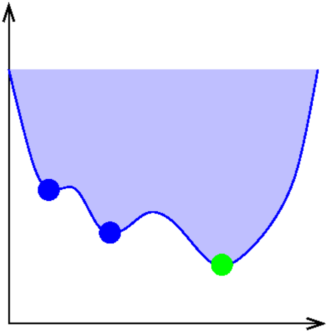
Challenges of Nonconvex MINLP

Definition (Local Minimum)

A point x^* is a local minimum of

$$\underset{x}{\text{minimize}} \ f(x) \quad \text{subject to } x \in \mathcal{F}$$

iff $\exists\, \mathcal{N}(x^*)$ such that $f(x) \geq f(x^*)$ for all $x \in \mathcal{N}(x^*) \cap \mathcal{F}$



Nonconvex $f(x)$ with three local and one global min

Challenges of Nonconvex MINLP

$$\underset{x}{\text{minimize}} \; f(x) \quad \text{subject to } c(x) \leq 0, \; x \in X, \; x_i \in \mathbb{Z} \; \forall \, i \in I$$

Definition (Global Minimum)

A point x^* is a global minimum of

$$\underset{x}{\text{minimize}} \; f(x) \quad \text{subject to } x \in \mathcal{F}$$

iff $f(x) \geq f(x^*)$ forall $x \in \mathcal{F}$

Remarks:

- NLP solvers are not guaranteed to find even local minima
  ... though they work remarkably well in practice!
- Global optimization is NP-hard (includes MIP: $(1 - x_i)x_i \leq 0$)
- Finding a global min is difficult ... proving it is really hard

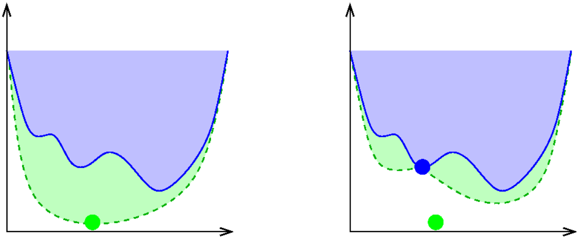
Outline

General Approach to Nonconvex MINLP

$$\underset{x}{\text{minimize}}\ f(x) \quad \text{subject to } c(x) \le 0,\ x \in X,\ x_i \in \mathbb{Z}\ \forall\, i \in I$$

Use our old MIP trick: convex relaxation!

- Relax integrality as before: $x_i \in \mathbb{R}\ \forall\, i \in I$
- Also need to relax $f(x)$ and constraints $c(x)$ ... new aspect
- Ensure relaxation is tractable: e.g. convex



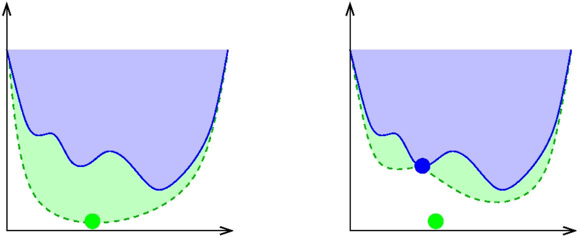
General Approach to Nonconvex MINLP

$$\underset{x}{\text{minimize}} \ f(x) \quad \text{subject to } c(x) \leq 0, \ x \in X, \ x_i \in \mathbb{Z} \ \forall \ i \in I$$

Relaxation provides lower bound, but solution infeasible in MINLP

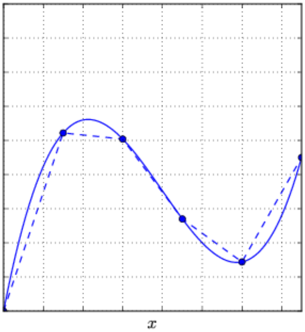
Need constraint enforcement to guarantee convergence

- Branching reduces area of relaxation
- Refinement tightens the relaxation over subdomain



Outline

Piecewise Linear Approximations for Univariate Functions



Consider univariate functions $g_i : \mathbb{R} \to \mathbb{R}$ and multivariate separable function

$$g(x) = \sum_{i=1}^{K} g_i(x_i)$$

Get approximation of $g(x)$ from approximations of $g_i(x_i)$

Two-step algorithm

- Obtain piecewise linear approximation
- Solve approx. problem as MILP & refine if necessary

Piecewise Linear Approximations for Univariate Functions

Given $g : [l, u] \to \mathbb{R}$, find piecewise linear $\hat{g} : [l, u]$ with $\hat{g}(x) \approx g(x)$ for all $x \in [l, u]$.

Consider d segments & breakpoints $l =: b^0 < b^1 < \cdots < b^d := u$ and function values $y^k = \hat{g}(b^k) = g(b^k)$, for $k = 0, 1, \ldots, d$

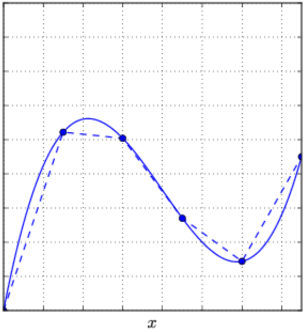
$$\hat{g}(x) = y^{k-1} + \left(\frac{y^k - y^{k-1}}{b^k - b^{k-1}} \right) (x - b^{k-1}), \; x \in [b^{k-1}, b^k], \; \forall k = 1, \ldots, d$$

Alternative definition: let $m_k = (y^k - y^{k-1})/(b^k - b^{k-1})$ slope of line segment then $a_k = y^k - m^k b^{k-1}$ is y-intercept

$$\Rightarrow \hat{g}(x) = a_k + m_k x, \; x \in [b^{k-1}, b^k], \; \forall k = 1, \ldots, d.$$

... now replace $g(x)$ by $\hat{g}(x)$ in MINLP

Piecewise Linear Approximations for Univariate Functions



Two competing aims:

1. $\min \|g(x) - \hat{g}(x)\|_{[l,u]}$
2. $\min \#$ breakpoints $= d$

Balance approximation error and solution time

Simplest approach: equidistant points ... better choice possible!

$y^k \neq g(b^k)$ can give better approximation

... we can formulate piecewise linear as MILP!

MILP Model (1) of Piecewise Linear Approximations

Given piecewise linear approximation of univariate $g(x) : \mathbb{R} \to \mathbb{R}$

$$g(x) \simeq \hat{g}(x) = a_k + m_k x, \;\; x \in [b^{k-1}, b^k], \;\; \forall k = 1, \ldots, d$$

Approach I: multiple choice model $\Rightarrow$ MILP

1. Introduce binary variables z_k, $k = 1, \ldots, d$,
   where $z_k = 1$ if $x \in [b^{k-1}, b^k]$; otherwise $z_k = 0$
2. Introduce variable w_k: $x = w_k$ in interval $[b^{k-1}, b^k]$
3. Add model equations to MINLP:

$$\sum_{k=1}^{d} w_k = x, \qquad \sum_{k=1}^{d} (m_k w_k + a_k z_k) = y, \quad \sum_{k=1}^{d} z_k = 1$$

$$b^{k-1} z_k \le w_k \le b^k z_k, \quad z_k \in \{0, 1\}, \;\; k = 1, \ldots, d$$

4. Replace $g(x)$ by y ... in MINLP model.

See [Jeroslow and Lowe, 1984] best for # breakpoints. $d \le 16$

MILP Model (2) of Piecewise Linear Approximations

Given piecewise linear approximation of univariate $g(x) : \mathbb{R} \to \mathbb{R}$

$$g(x) \simeq \hat{g}(x) = a_k + m_k x, \;\; x \in [b^{k-1}, b^k], \;\; \forall k = 1, \ldots, d$$

Approach 2: convex combination model $\Rightarrow$ MILP

1. Introduce binary variables $z_k = 1$ iff $x \in [b^{k-1}, b^k]$
2. Introduce continuous variable λ_k convex combination
3. Add model equations to MINLP ... related to SOS-2

$$\sum_{k=0}^{d} \lambda_k b^k = x, \qquad \sum_{k=0}^{d} \lambda_k y^k = y,$$

$$\sum_{j=k}^{d} \lambda_j \le \sum_{j=k}^{d} z_j, \qquad \sum_{j=0}^{k-1} \lambda_j \le \sum_{j=1}^{k} z_j, \quad k = 1, \ldots, d,$$

$$\sum_{k=0}^{d} \lambda_k = 1 \qquad \sum_{k=1}^{d} z_k = 1,$$

$$\lambda_k \ge 0, \quad k = 0, 1, \ldots, d \quad z_k \in \{0, 1\}, \quad k = 1, \ldots, d.$$

SOS-2 Model of Piecewise Linear Approximations

Given piecewise linear approximation of univariate $g(x) : \mathbb{R} \to \mathbb{R}$

$$g(x) \simeq \hat{g}(x) = a_k + m_k x, \;\; x \in [b^{k-1}, b^k], \;\; \forall k = 1, \ldots, d$$

Model piecewise linear as SOS-2 without additional variables!

Definition (SOS-2 Sets)

Set of variables $\lambda = (\lambda_0, \lambda_1, \ldots, \lambda_d)$ is SOS-2, iff at most two adjacent λ_i nonzero.

Gives formulation (related to MILP Model (2) above)

$$\sum_{k=0}^{d} \lambda_k b^k = x, \qquad \sum_{k=0}^{d} \lambda_k y^k = y,$$

$$\sum_{k=0}^{d} \lambda_k = 1,$$

$$\lambda_k \geq 0, \quad k = 0, 1, \ldots, d \quad (\lambda_0, \lambda_1, \ldots, \lambda_d) \text{ is SOS2.}$$

Implemented in most MILP solvers

SOS-2 Model of Piecewise Linear Approximations

How can we branch on SOS-2 set?

$$\sum_{k=0}^{d} \lambda_k b^k = x, \quad \sum_{k=0}^{d} \lambda_k y^k = y, \quad \sum_{k=0}^{d} \lambda_k = 1,$$

and $\lambda_k \geq 0, \quad k = 0, 1, \ldots, d \quad (\lambda_0, \lambda_1, \ldots, \lambda_d)$ is SOS2

If solution $\hat{\lambda}$ of relaxation violates SOS-2 condition the

1. Select index $k \in \{1, \ldots, d\}$ such that: $\exists j_1 < k$ with $\lambda_{j_1} > 0$ and $\exists j_2 > k$ with $\lambda_{j_2} > 0$
2. Create two branches:
   1. Branch 1 set $\lambda_j = 0$ for all $j < k$
   2. Branch 2 set $\lambda_j = 0$ for all $j > k$

See [Beale and Tomlin, 1970]; generalizes to multivariate $g(x)$

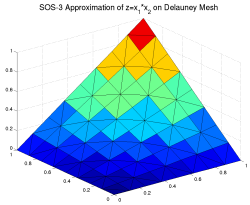
... more models in paper

Outline

Piecewise Linear Approximations for Multivariate Functions

SOS-2 generalizes to multiple dimensions [Beale and Tomlin, 1970]

- Multivariate $g : \mathbb{R}^d \to \mathbb{R}$
- Piecewise linear approx. of $g(x)$
- Choose breakpoints b^k, $k = 1, \ldots, q$
- Partition $\otimes_{i=1}^{d} [l_i, u_i]$ into simplices
- Approximation $\hat{g}(x)$ with $\lambda_k \geq 0$



$$\hat{g}(x) = \sum_{k=1}^{q} \lambda_k g(b^k), \quad x = \sum_{k=1}^{q} \lambda_k b^k, \quad 1 = \sum_{k=1}^{q} \lambda_k$$

Definition (SOS-$\{d+1\}$ Set Condition)

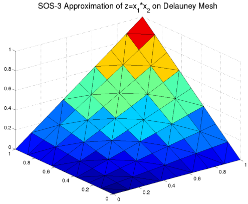
The set $(\lambda_1, \ldots, \lambda_q)$ satisfies SOS-$\{d+1\}$ condition, iff at most $d+1$ λ_k non-zero on single simplex

Piecewise Linear Approximations for Multivariate Functions

Example: Approximation of 2D function $u = g(v, w)$

Triangularization of $[v_L, v_U] \times [w_L, w_U]$ domain

1. $v_L = v_1 < \ldots < v_k = v_U$
2. $w_L = w_1 < \ldots < w_l = w_U$
3. function $u_{ij} := g(v_i, w_j)$
4. λ_{ij} weight of vertex (i, j)

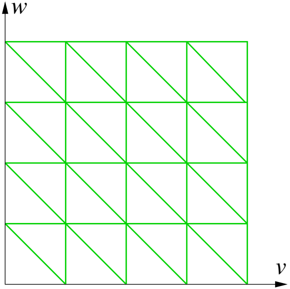


Piecewise Linear Approximations for Multivariate Functions

Example: Approximation of 2D function $u = g(v, w)$

Triangularization of $[v_L, v_U] \times [w_L, w_U]$ domain

1. $v_L = v_1 < \ldots < v_k = v_U$
2. $w_L = w_1 < \ldots < w_l = w_U$
3. function $u_{ij} := g(v_i, w_j)$
4. λ_{ij} weight of vertex (i, j)

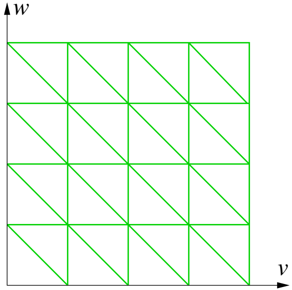


Piecewise Linear Approximations for Multivariate Functions

Example: Approximation of 2D function $u = g(v, w)$

Triangularization of $[v_L, v_U] \times [w_L, w_U]$ domain

1. $v_L = v_1 < \ldots < v_k = v_U$
2. $w_L = w_1 < \ldots < w_l = w_U$
3. function $u_{ij} := g(v_i, w_j)$
4. λ_{ij} weight of vertex (i, j)



$$v = \sum_{i=1}^{k} \lambda_{ij} v_i, \quad w = \sum_{j=1}^{l} \lambda_{ij} w_j, \quad u = \sum_{i=1}^{k} \sum_{j=1}^{l} \lambda_{ij} u_{ij}, \quad \lambda_{ij} \geq 0$$

$1 = \sum \lambda_{ij}$ is SOS3 ...

Piecewise Linear Approximations for Multivariate Functions

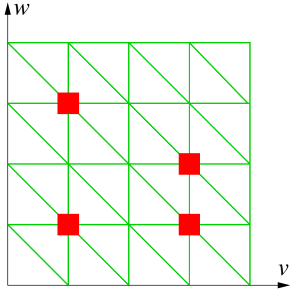
SOS3: $\sum \lambda_{ij} = 1$ & set condition holds

1. $v = \sum \lambda_{ij} v_i$ ... convex combinations
2. $w = \sum \lambda_{ij} w_j$
3. $u = \sum \lambda_{ij} u_{ij}$

$\{\lambda_{11}, \ldots, \lambda_{kl}\}$ satisfies set condition

$\Leftrightarrow \; \exists$ triangle $\Delta : \{(i,j) : \lambda_{ij} > 0\} \subset \Delta$

i.e. nonzeros in single triangle Δ

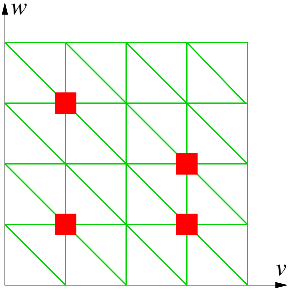


violates set condn

Piecewise Linear Approximations for Multivariate Functions

Branching on SOS3 when λ violates set condition

- compute centers:
  $\hat{v} = \sum \lambda_{ij} v_i$ &
  $\hat{w} = \sum \lambda_{ij} w_i$
- find s, t such that
  $v_s \leq \hat{v} < v_{s+1}$ &
  $w_t \leq \hat{w} < w_{t+1}$
- branch on v or w



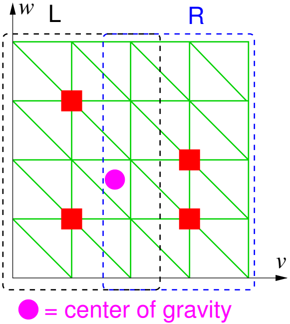
violates set condn

Branching on SOS3

Piecewise Linear Approximations for Multivariate Functions

Branching on SOS3 when λ violates set condition

- compute centers: $\hat{v} = \sum \lambda_{ij} v_i$ & $\hat{w} = \sum \lambda_{ij} w_i$
- find s, t such that $v_s \leq \hat{v} < v_{s+1}$ & $w_t \leq \hat{w} < w_{t+1}$
- branch on v or w



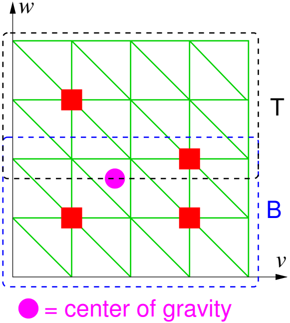
Branching on SOS3

vertical branching: $\sum_L \lambda_{ij} = 1$ $\quad \sum_R \lambda_{ij} = 1$

Piecewise Linear Approximations for Multivariate Functions

Branching on SOS3 when λ violates set condition

- compute centers:
  $\hat{v} = \sum \lambda_{ij} v_i$ &
  $\hat{w} = \sum \lambda_{ij} w_i$
- find s, t such that
  $v_s \leq \hat{v} < v_{s+1}$ &
  $w_t \leq \hat{w} < w_{t+1}$
- branch on v or w

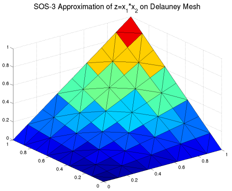


Branching on SOS3

horizontal branching: $\sum_T \lambda_{ij} = 1 \quad \sum_B \lambda_{ij} = 1$

Piecewise Linear Approximations for Multivariate Functions

Pitfall: Exponential Complexity of SOS



- Approximate $g(x)$ for $x \in \mathbb{R}^n$
- Use p breakpoints in each dimension
  $\Rightarrow$ $\Rightarrow$ p^n SOS-variables λ_i

e.g. expression for real power has $n = 8$ variables ... impractical

... use decomposition of functions, see [Kesavan et al., 2004]

Remedy: Decomposition of Nonlinear Functions

SOS-approximation needs p^n SOS-variables λ_k

Idea: decompose $h(x)$ into simpler functions:

$$\begin{aligned} w_j &= x_j & j &= 1, \ldots, s, \\ w_{s+j} &= g_j(w_{j_1}\{, w_{j_2}\}) & j &= 1, \ldots, K, \\ h(x, y) &= w_{s+t+K}, \end{aligned}$$

where g_j are univariate or bivariate and $j_1, j_2 < s + t + j$

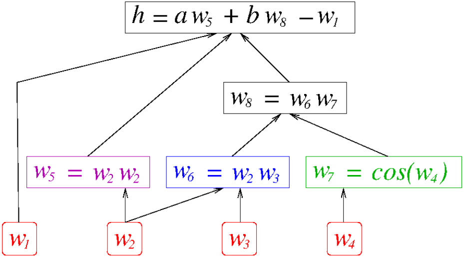
Remedy: Decomposition of Nonlinear Functions

Consider

$$g(x_1, x_2, x_3, x_4) = ax_2^2 + bx_2x_3\cos(x_4) - x_1$$

where a and b constants.

- $w_j = x_j \quad j = 1, \ldots,$
- $w_5 = w_2^2$
- $w_6 = w_2 w_3$
- $w_7 = \cos(w_4)$
- $w_8 = w_6 w_7$
- $g = aw_5 + bw_8 - w_1$



Decomposition not unique: e.g. $w_6 = \cos(w_4)$ etc.

Remedy: Decomposition of Nonlinear Functions

Example: Expression for active power

$$P_{ij} = \nu_i^2 \, (y_{ij} \cos(\zeta_{ij}) + g_{ij}) - \nu_i \nu_j y_{ij} \sin(\zeta_{ij} + \theta_i - \theta_j)$$

Simple functions:

- ν_i^2
- $\cos(\zeta_{ij})$
- $\sin(w_{j_1})$, where $w_{j_1} = \zeta_{ij} + \theta_i - \theta_j$
- 5 bilinear terms like $\nu_i \nu_j$

$\Rightarrow$ need only $5p^2 + 3p$ SOS variables, λ ... much smaller p^8

Decomposition Nonconvex MINLP

Consider MINLP in format

$$(P) \begin{cases} \underset{x}{\text{minimize}} & g_0(x), \\ \text{subject to} & g_i(x) \leq 0, \; i = 1, .., m, \\ & x \in X, \; x_I \in \mathbb{Z}^p \end{cases}$$

... and assume that it is factorable

Definition (Factorable MINLP)

A MINLP is factorable if every function can be written as a sum of products of unary functions.

Decomposition Nonconvex MINLP

$$(P)\left\{\min_x\ g_0(x), \quad \text{s.t. } g_i(x) \le 0,\ i=1,..,m, \quad x \in X,\ x_I \in \mathbb{Z}^p\right.$$

Introduce variables w, write MINLP (P) equivalently as

$$(D)\left\{\begin{array}{lll}\underset{x,w}{\text{minimize}} & w_{0,K_0} & \\ \text{subject to} & w_{ij} = x_j & \forall i,j \\ & w_{i,n+j} = g_{ij}(w_{i,j_1}\{, w_{i,j_2}\}) & \forall i,j \\ & w_{i,n+K_i} \le 0 & i=1,\ldots,m \\ & x \in X,\ x_i \in \mathbb{Z} & \forall i \in I\end{array}\right.$$

...equivalent to MINLP (P) ... related to automatic differentiation
where $g_{ij}(w_{i,j_1}\{, w_{i,j_2}\})$ univariate/bivariate component of $c_i(x)$

Basis of general approach to nonconvex MINLP!

Example: Decomposition of Nonlinear Functions

Example: Expression for active power is factorable

$$P_{ij} = \nu_i^2\ (y_{ij}\cos(\zeta_{ij}) + g_{ij}) - \nu_i\nu_j y_{ij}\sin(\zeta_{ij} + \theta_i - \theta_j)$$

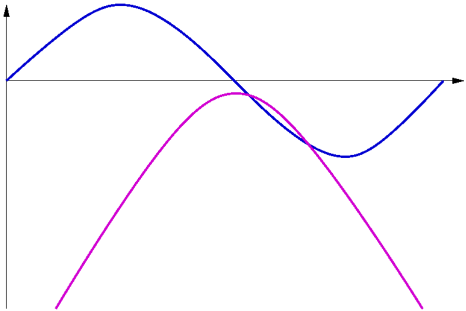
Get factorable form:

$$\begin{aligned}
&w_{11} = \nu_i, && w_{21} = \nu_j, && w_{31} = \zeta_{ij} \\
&w_{12} = w_{11}^2 && w_{22} = w_{11}w_{21} && w_{32} = \cos(w_{31}) \\
&w_{33} = \cos(w_{31} + \theta_i - \theta_j), && w_{34} = w_{33}w_{22}, && w_{35} = y_{ij}w_{32} + g_{ij} \\
&w_{36} = w_{35}w_{12}
\end{aligned}$$

BARON & Couenne solvers use factorable format.

Outline

SOS Approximations Become Infeasible



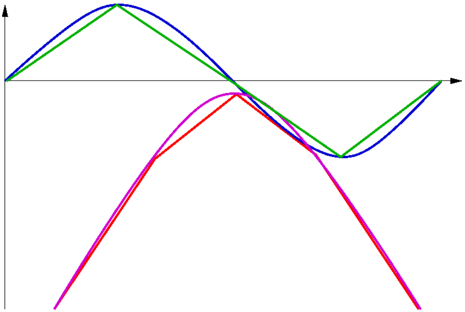
SOS approx infeasible

$\sin(x) = 0$

$-0.35\,(x - \pi)^2 - 0.3 = 0$

...observed infeasible SOS on some power-grid examples!

SOS Approximations Become Infeasible

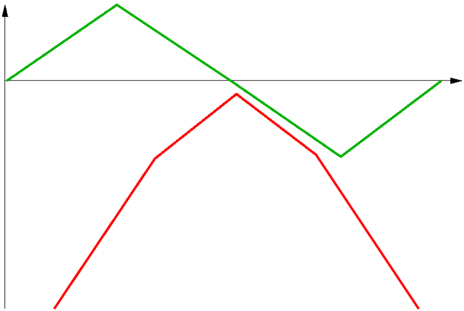


SOS approx infeasible

$\sin(x) = 0$
$-0.35\ (x - \pi)^2 - 0.3 = 0$

. . . observed infeasible SOS on some power-grid examples!

SOS Approximations Become Infeasible



SOS approx infeasible

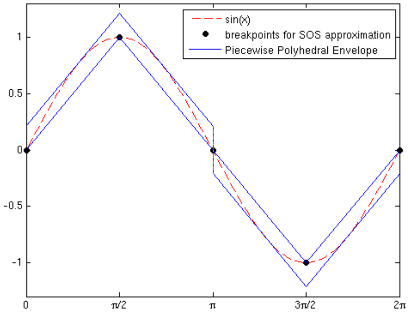
$\sin(x) = 0$

$-0.35\ (x - \pi)^2 - 0.3 = 0$

…observed infeasible SOS on some power-grid examples!

Remedy: Piecewise Polyhedral Envelopes

Idea: Outer approximation by piecewise polyhedral envelopes



Univariate $w_g = g(w)$ represented by envelope:

$$\sum_{k \in I} \lambda_k \left(g(w_k) - L_k \right) \leq w_g \leq \sum_{k \in I} \lambda_k \left(g(w_k) + U_k \right)$$

Remedy: Piecewise Polyhedral Envelopes

Obtain bound L_k by solving

$$L_k = \max_{w \in [w^k, w^{k+1}], \lambda^k + \lambda^{k+1} = 1} \left(0, \lambda^k g(w^k) + \lambda^{k+1} g(w^{k+1}) - g(w)\right)$$

... similar for U_k

Bounds L_k, U_k pre-computed on $[w_k, w_{k+1}]$, e.g. $g(w) = w^2$:

$$L_k = (w_{k+1} - w_k)^2/4, \qquad U_k = 0$$

See Emilie's thesis for other functions ...

Piecewise Polyhedral Envelopes for $g = x\ y$

Theorem: Every (x, y, xy) with $l_x \leq x \leq u_x$ and $l_y \leq y \leq u_y$ is unique convex combination of $(l_x, l_y, l_x l_y)$, $(l_x, u_y, l_x u_y)$, $(u_x, l_y, u_x l_y)$ and $(u_x, u_y, u_x u_y)$, i.e. $\exists \lambda_i \geq 0,\ i = 1, \ldots, 4$:

$$\begin{pmatrix} x \\ y \\ xy \\ 1 \end{pmatrix} = \begin{bmatrix} l_x & l_x & u_x & u_x \\ l_y & u_y & l_y & u_y \\ l_x l_y & l_x u_y & u_x l_y & u_x u_y \\ 1 & 1 & 1 & 1 \end{bmatrix} \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \end{pmatrix}$$

Implies $L_k = U_k = 0$, and equality (tighter relaxation):

$$w_{xy} = \sum_{(i,j) \in I} \lambda_{ij} x_i y_j$$

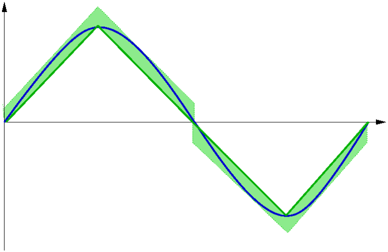
Piecewise Envelope Problem

Proposition: (E) is an outer approximation of (D) and hence (P).

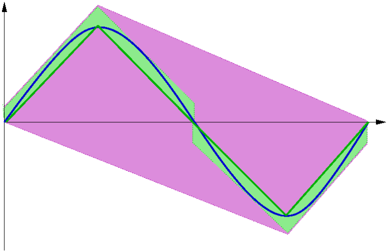
$$(E)\begin{cases} \underset{x,w,\lambda}{\text{minimize}} & w_{0,K_0} \\ \text{subject to} & w_{ij} = x_j, \\ & x_j = \sum_{k\in I_j} \lambda_{j_k} x_{j_k}, \qquad 1 = \sum_{k\in I_j} \lambda_{j_k} \\ & w_{i,n+j} \geq \sum_{k\in I_{ij}} \lambda_{ij}^k \left(g_{ij}(w_{i,j_1}^k \{, w_{i,j_2}^k\}) - L_{ijk} \right) \\ & w_{i,n+j} \leq \sum_{k\in I_{ij}} \lambda_{ij}^k \left(g_{ij}(w_{i,j_1}^k \{, w_{i,j_2}^k\}) + U_{ijk} \right) \\ & w_{i,s+K_i} = 0 \\ & x \in X,\ x_I \mathbb{Z}^p, \text{ and } w \in W, \end{cases}$$

where W deduced from x bounds;
and blue part replaces $w_{i,n+j} = g_{ij}(w_{i,j_1}^k \{, w_{i,j_2}^k\})$

Piecewise Envelope Problem: Illustration



SOS Outer Approximation



Convex Hull

Outline

Branch-and-Refine: Outline

Classical Branch-and-Bound:

Solve envelope problem (E) branch on SOS-condition or $x_I \in \mathbb{Z}^p$

$\Rightarrow$ large discretization error or large number of λ_k variables

Idea: Instead refine discretization after branching:

- tighten envelope as we go down tree: refine
- exploit exactness of bilinear terms $w_1 w_2$
- better numerical results

Branch-and-Refine: Branching

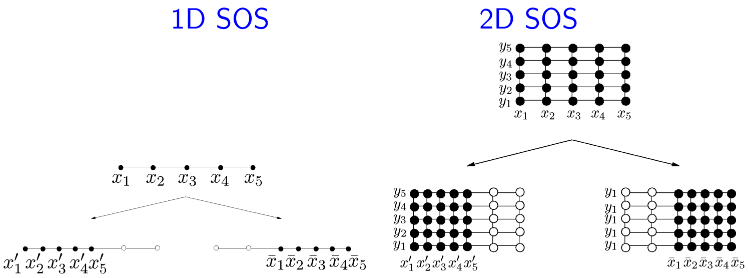


Illustration of branching and refinement

Branch-and-Refine: Fathoming Rules

Also solve NLP(X_k):

$$\begin{cases} z_{\text{NLP}_k} := \underset{x}{\text{minimize}}\ g_0(x) \\ \text{subject to} \quad g_i(x) = 0,\ i = 1, .., m \\ \qquad\qquad x \in X_k, \end{cases}$$

... upper bound on node (X_k).

Fathoming Rules:

1. infeasible LP relaxation
2. NLP(X_k) solution same as LP(X_k) relaxation
3. LP relaxation dominated by incumbent

Branch-and-Refine: Algorithm

set $U = \infty$, $k = 1$ & put LP(X_k) on stack
while stack is not empty
  solve LP(X_k) ... solution x^k
  if LP(X_k) infeasible or $z_{\text{LP}_k} \geq U - \epsilon$ **then**
    fathom node (case 1. or 3.)
  else
    solve NLP(X_k) ... solution $\hat{x}^k$
    if $z_{\text{NLP}_k} < U - \epsilon$ & $\hat{x}_I^k$ integer **then**
      update $U := z_{\text{NLP}_k}$ & incumbent $x^* := \hat{x}$
    if $\left| z_{\text{NLP}_k} - z_{\text{LP}_k} \right| \leq \epsilon$ **then**
      fathom node (case 2.)
    else
      branch creating two new LPs

Theorem: If $x \in X$ is bounded $\Rightarrow$ get ϵ-optimal solution.

Test Problems (Generic)

prob	#var	#cons	#var OA	#cons OA	#sets λ	#disc
pb0	4	2	44	32	6	1
pb1	4	2	44	32	6	1
pb2	6	2	41	30	5	1
pb3	6	2	41	30	5	1
pb4	12	4	97	71	11	2
pb5	12	4	97	71	11	2
pb6	12	4	143	97	19	3
pb7	12	4	143	97	19	3
pb8	12	4	119	77	14	2
pb9	12	4	119	77	14	2
pb10	10	4	111	72	13	2
pb11	10	4	111	72	13	2
pb12	24	8	275	187	40	6
pb13	24	8	275	187	40	6

Test Problems (Tertiary Voltage Control)

prob	#var	#cons	#var OA	#cons OA	#sets λ	#disc
TVC1	16	9	269	200	39	6
TVC2	18	9	275	204	40	6
TVC3	27	15	422	315	61	9
TVC4	27	15	422	315	61	9
TVC5	37	21	602	449	87	13
TVC6	38	21	635	472	92	14

... moderately sized problems

Complexity of nonconvex MINLPs depends on
terms in computational graph $\simeq$ #sets λ

Do We Need Global Solvers?

Comparison with NLP solvers

solver	# Problems Solved	# Global Solutions
BnR	20	20
Filter	12	8
IPOPT	17	14
KNITRO	17	13

Comparison with MINLP solvers

solver	# Problems Solved	# Global Solutions
BnR	20	20
BONMIN	15	11
MINLPBB	11	9

Implementation Details & Tricks

- LPs solved with CPLEX
- Decomposition hand-coded by Emilie (yikes!)
  - exploit common sub-expressions
  - Can be automated, similar to automatic differentiation (AD)
  - Modern global solvers do this automatically
- NLPs solved with FilterSQP (AD for gradients/Hessian)
- Propagate & strengthen bounds through computational graph
- Pre-solve (LP) to reduce range of variables (like BARON)
  - Adaptive presolve is best: tail-off factor
- Pseudo-cost branching (generalized to nonconvex)
- Best-estimate node selection (generalized to nonconvex)

Numerical Results (# LPs solved)

prob	basic	+presolve	+var-select	+node-select
pb0	63	63	68	68
pb1	133	131	79	68
pb2	2115	3237	194	260
pb3	135	197	121	97
pb4	15389	11388	120	120
pb5	3009	257	145	145
pb6	65800	6145	348	292
pb7	377	1353	1235	1121
pb8	fail	198817	263	241
pb9	62149	33668	442	442
pb10	113846	51816	205	197
pb11	3806	7349	558	258
pb12	fail	33407	1503	1056
pb13	fail	8093	17388	3885

Numerical Results (# LPs solved)

prob	basic	+presolve	+var-select	+node-select
TVC1	108861	40446	7756	8031
TVC2	fail	72270	5792	5547
TVC3	62045	861	627	627
TVC4	fail	38792	1396	1582
TVC5	fail	7369	5619	4338
TVC6	fail	12131	6096	5503

Outline

Summary and Student Discussion

Key Points

- Nonconvex functions make MINLPs much harder
- General approach based on underestimators
- Piecewise linear functions & factorable functions

Short Presentations by Students Volunteers:

- Sebastien Mathieu, University of Liège
- Azamat Shakhimardanov, KU Leuven
- Lin Zhang, KU Leuven
- Yansong Guo, KU Leuven
- David Jalúvka, KU Leuven
- Joly Arnaud, University of Liège
- Damien Gerard, University of Liège

Office Hours: Wednesday after the course in room 115

Atamtürk, A. and Narayanan, V. (2010).
Conic mixed-integer rounding cuts.
Mathematical Programming A, 122(1):1–20.

Beale, E. and Tomlin, J. (1970).
Special facilities in a general mathematical programming system for non- convex problems using ordered sets of variables.
In Lawrence, J., editor, *Proceedings of the 5th International Conference on Operations Research*, pages 447–454, Venice, Italy.

Çezik, M. T. and Iyengar, G. (2005).
Cuts for mixed 0-1 conic programming.
Mathematical Programming, 104:179–202.

Drewes, S. (2009).
Mixed Integer Second Order Cone Programming.
PhD thesis, Technische Universität Darmstadt.

Drewes, S. and Ulbrich, S. (2012).
Subgradient based outer approximation for mixed integer second order cone programming.
In *Mixed Integer Nonlinear Programming*, volume 154 of *The IMA Volumes in Mathematics and its Applications*, pages 41–59. Springer, New York.
ISBN 978-1-4614-1926-6.

Jeroslow, R. and Lowe, J. (1984).
Modelling with integer variables.
Mathematical Programming Studies, 22:167–84.

Kesavan, P., Allgor, R. J., Gatzke, E. P., and Barton, P. I. (2004).
Outer approximation algorithms for separable nonconvex mixed-integer nonlinear programs.
Mathematical Programming, 100:517–535.

Stubbs, R. and Mehrotra, S. (1999).
A branch-and-cut method for 0-1 mixed convex programming.
Mathematical Programming, 86:515–532.