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

1. MISOCPP Cuts

2. Global Optimization of Nonconvex MINLP
   - Challenges of Nonconvex MINLP
   - General Approach to Nonconvex MINLP

3. Piecewise Linear Approach to Nonconvex MINLP
   - Piecewise Linear Approach to Univariate Nonconvex MINLP
   - Piecewise Linear Approach to Multivariate Nonconvex MINLP
   - Beyond Piecewise Linear Functions
   - A Branch-and-Refine Algorithm

4. Summary and Student Discussion
Outline

1. MISOCPL Cuts

2. Global Optimization of Nonconvex MINLP
   - Challenges of Nonconvex MINLP
   - General Approach to Nonconvex MINLP

3. Piecewise Linear Approach to Nonconvex MINLP
   - Piecewise Linear Approach to Univariate Nonconvex MINLP
   - Piecewise Linear Approach to Multivariate Nonconvex MINLP
   - Beyond Piecewise Linear Functions
   - A Branch-and-Refine Algorithm

4. Summary and Student Discussion
The Chvátal-Gomory Procedure

A general procedure for generating valid inequalities for ILP

\[
\begin{aligned}
\text{minimize} & \quad c^T x \\
\text{subject to} & \quad Ax \leq b, \ x \geq 0, \ x \in \mathbb{Z}^n
\end{aligned}
\]

- Let the columns of \( A \in \mathbb{R}^{m \times n} \) be denoted by \( \{a_1, a_2, \ldots, a_n\} \)
- \( S = \{x \in \mathbb{Z}_+^n \mid Ax \leq b\} \) feasible set of ILP.

1. Choose nonnegative multipliers \( u \in \mathbb{R}_+^m \)
2. \( u^T Ax \leq u^T b \) is a valid inequality: \( \sum_{j \in N} u_j a_j x_j \leq u^T b \).

3. \( \sum_{j \in N} \lfloor u^T a_j \rfloor x_j \leq u^T b \), since \( x \geq 0 \).

4. \( \sum_{j \in N} \lfloor u^T a_j \rfloor x_j \leq \lfloor u^T b \rfloor \) 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

\[
\begin{align*}
\text{minimize} & \quad c^T x \\
\text{subject to} & \quad Ax \leq b, \ x \geq 0, \ x \in \mathbb{Z}^n
\end{align*}
\]

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

\[
\begin{align*}
\text{minimize} & \quad f^T x \\
\text{subject to} & \quad Ax \succeq_K b \\
& \quad x_I \in \{0, 1\}^p, \ 0 \leq x \leq U
\end{align*}
\]

- $\mathcal{K}$: Homogeneous, self-dual, proper, convex cone
- $x \succeq_K x' \iff (x - x') \in \mathcal{K}$
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**: $\mathcal{K}_s = \{x = \text{vec}(X) \mid X = X^T, X \text{ positive semi-definite}\}$
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**: $\mathcal{K}_s = \{x = \text{vec}(X) \mid X = X^T, X \text{ positive semi-definite}\}$
- **Dual Cone**: $\mathcal{K}^* := \{u \mid u^T z \geq 0 \ \forall z \in \mathcal{K}\}$
- **Extension** is clear from the following equivalence:

\[
Az \succeq_{\mathcal{K}} b \iff u^T Az \geq u^T b \ \forall u \succeq_{\mathcal{K}^*} 0
\]
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:

\[ Az \succeq_\mathcal{K} b \iff u^T Az \geq u^T b \ \forall u \succeq_{\mathcal{K}^*} 0 \]

- 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:

\[
\begin{align*}
\text{minimize} & \quad c^T x \\
\text{subject to} & \quad x \in \mathcal{K} \\
& \quad Ax = b, \quad l \leq x \leq u \\
& \quad x_i \in \mathbb{Z} \quad \forall i \in I.
\end{align*}
\]

\(x \in \mathcal{K}\) 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:

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

and define outer approximations from subgradients of $\|x_j^1\|_2 = x_{j0}$:

$$J_a(\bar{x}) := \{j : g_j(\bar{x}) = 0, \ \bar{x} \neq 0\}, \quad \text{active different.}$$

$$J_{0+}(\bar{x}, \bar{s}) := \{j : \bar{x}_j = 0, \ \bar{s}_{j0} > 0\}, \quad \text{strongly active}$$

$$J_{00}(\bar{x}, \bar{s}) := \{j : \bar{x}_j = 0, \ \bar{s}_{j0} = 0\}, \quad \text{weakly active}$$

... and derive OA master problem $(g_j(\bar{x}) = \|x_j^1\|_2 - x_{j0})$
Outer Approximation for MISOCPs

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

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

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

Convergence, see [Drewes and Ulbrich, 2012]

... Exercise: Is this OA approach finite?
Theorem ([Drewes, 2009])

Continuous SOCP & dual satisfy Slater’s CQ & $l_l \geq 0$.

$\bar{x}$ with $\bar{x}_l \notin \mathbb{Z}^p$ solution of SOCP($x_l^{(k)}$), $(\bar{s}, \bar{y})$ dual.

Then following cut is valid for MISOCP,

$$\lceil (A_l^T (\bar{y} - \Delta y) \bar{s}_l) \rceil s_l \geq \lceil (\bar{y} - \Delta y) b \rceil,$$

where $\Delta y$ solves

$$\begin{pmatrix} -A_c \\ A_l \end{pmatrix} \Delta y = \begin{pmatrix} c_c \\ 0 \end{pmatrix}.$$  

If $\lceil (\bar{y} - \Delta y) b \rceil \notin \mathbb{Z}$, then cut off $\bar{x}$. 
Example of Gomory for MISOCP

Example:

\[
\begin{align*}
\min_{x} & \quad -x_2 \\
\text{s.t.} \quad & -3x_2 + x_3 \leq 0 \\
& \quad 2x_2 + x_3 \leq 3 \\
& \quad 0 \leq x_1, x_2 \leq 3 \\
& \quad x_1 \geq \| (x_2, x_3)^T \|_2 \\
& \quad x_1, x_2 \in \mathbb{Z},
\end{align*}
\]

relaxed solution: \( (3, \frac{12}{5}, -\frac{9}{5}) \).

The Gomory cut \( x_2 \leq 2 \)
Other Work on MISOCPP

Related work on MISOCPP (simplest generalization of MILP)

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

1. MISOCPCuts

2. Global Optimization of Nonconvex MINLP
   - Challenges of Nonconvex MINLP
   - General Approach to Nonconvex MINLP

3. Piecewise Linear Approach to Nonconvex MINLP
   - Piecewise Linear Approach to Univariate Nonconvex MINLP
   - Piecewise Linear Approach to Multivariate Nonconvex MINLP
   - Beyond Piecewise Linear Functions
   - A Branch-and-Refine Algorithm

4. Summary and Student Discussion
Challenges of Nonconvex MINLP

Mixed-Integer Nonlinear Program (MINLP)

\[
\begin{align*}
\min _{x} & \quad f(x) \\
\text{subject to} & \quad c(x) \leq 0, \ x \in X, \ x_i \in \mathbb{Z} \ \forall \ i \in I
\end{align*}
\]

... 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 \[
\left\{ x \mid c(x) \leq 0, \ x \in X \right\}
\]

... can be disjoint, may have no interior
Challenges of Nonconvex MINLP

Mixed-Integer Nonlinear Program (MINLP)

\[ \minimize_x f(x) \text{ subject to } c(x) \leq 0, \; x \in X, \; x_i \in \mathbb{Z} \; \forall \; i \in I \]

Nonconvexity arise naturally

- Take nonlinear, convex \( c(x) \) and consider \( l \leq c(x) \leq u \)
  \( \Rightarrow \) nonconvex feasible region, e.g. \( \{1 \leq x_1^2 + x_2^2 \leq 2\} \)

- Nonlinear equations arise naturally in power grid applications
  e.g. nonlinear (AC) power flow model:

\[
F(U_k, U_l, \theta_k, \theta_l) := b_{kl} U_k U_l \sin(\theta_k - \theta_l) + g_{kl} U_k^2 \\
- g_{kl} U_k U_l \cos(\theta_k - \theta_l)
\]

- 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)

\[
\begin{align*}
\text{minimize } & \quad f(x) \\
\text{subject to } & \quad c(x) \leq 0, \ x \in X, \ x_i \in \mathbb{Z} \ \forall \ i \in I
\end{align*}
\]

**Definition (Convexity)**

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

\[
f(x^{(1)}) \geq f(x^{(0)}) + (x^{(1)} - x^{(0)})^T \nabla f^{(0)}
\]

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)

\[
\min_{x} f(x) \quad \text{subject to } c(x) \leq 0, \ x \in X, \ x_i \in \mathbb{Z} \ \forall \ i \in I
\]

Definition (Convexity)

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

\[
f(x^{(1)}) \geq f(x^{(0)}) + (x^{(1)} - x^{(0)})^T \nabla f^{(0)}
\]

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

\[
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{subject to} & \quad x \in \mathcal{F}
\end{align*}
\]

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 \textit{local} and one \textit{global} min
Challenges of Nonconvex MINLP

\[
\min_{\mathbf{x}} f(\mathbf{x}) \quad \text{subject to } c(\mathbf{x}) \leq 0, \ x \in X, \ x_i \in \mathbb{Z} \ \forall \ i \in I
\]

**Definition (Global Minimum)**

A point \( x^* \) is a global minimum of

\[
\min_{\mathbf{x}} f(\mathbf{x}) \quad \text{subject to } \mathbf{x} \in \mathcal{F}
\]

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

**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

1. MISOCP Cuts

2. Global Optimization of Nonconvex MINLP
   - Challenges of Nonconvex MINLP
   - General Approach to Nonconvex MINLP

3. Piecewise Linear Approach to Nonconvex MINLP
   - Piecewise Linear Approach to Univariate Nonconvex MINLP
   - Piecewise Linear Approach to Multivariate Nonconvex MINLP
   - Beyond Piecewise Linear Functions
   - A Branch-and-Refine Algorithm

4. Summary and Student Discussion
General Approach to Nonconvex MINLP

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

Use our old MIP trick: \textit{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. \textit{convex}
General Approach to Nonconvex MINLP

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

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

1. MISOCP Cuts

2. Global Optimization of Nonconvex MINLP
   - Challenges of Nonconvex MINLP
   - General Approach to Nonconvex MINLP

3. Piecewise Linear Approach to Nonconvex MINLP
   - Piecewise Linear Approach to Univariate Nonconvex MINLP
   - Piecewise Linear Approach to Multivariate Nonconvex MINLP
   - Beyond Piecewise Linear Functions
   - A Branch-and-Refine Algorithm

4. Summary and Student Discussion
Consider univariate functions $g_i : \mathbb{R} \rightarrow \mathbb{R}$ 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
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 \( k = 0, 1, \ldots, d \)

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

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

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

... now replace \( g(x) \) by \( \hat{g}(x) \) in MINLP
Two competing aims:

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

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 \( g(x) : \mathbb{R} \rightarrow \mathbb{R} \)

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

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:

\[
\begin{align*}
\sum_{k=1}^{d} w_k &= x, \\
\sum_{k=1}^{d} (m_k w_k + a_k z_k) &= y, \\
\sum_{k=1}^{d} z_k &= 1 \\
\end{align*}
\]

\[
b^{k-1} z_k \leq w_k \leq b^k z_k, \quad z_k \in \{0, 1\}, \quad k = 1, \ldots, d
\]

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

See [Jeroslow and Lowe, 1984] best for \# breakpoints. \( d \leq 16 \)
MILP Model (2) of Piecewise Linear Approximations

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

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

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 $\lambda_k$ convex combination
3. Add model equations to MINLP ... related to SOS-2

\[
\begin{align*}
\sum_{k=0}^{d} \lambda_k b^k &= x, \\
\sum_{j=k}^{d} \lambda_j &\leq \sum_{j=k}^{d} z_j, \\
\sum_{k=0}^{d} \lambda_k &= 1 \\
\lambda_k &\geq 0, \quad k = 0, 1, \ldots, d
\end{align*}
\]

\[
\begin{align*}
\sum_{k=0}^{d} \lambda_k y^k &= y, \\
\sum_{j=0}^{k-1} \lambda_j &\leq \sum_{j=1}^{k} z_j, \quad k = 1, \ldots, d, \\
\sum_{k=1}^{d} z_k &= 1, \\
z_k &\in \{0, 1\}, \quad k = 1, \ldots, d.
\end{align*}
\]
SOS-2 Model of Piecewise Linear Approximations

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

\[
g(x) \approx \hat{g}(x) = a_k + m_k x, \quad x \in [b_k^{-1}, b_k], \quad \forall k = 1, \ldots, d
\]

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 \( \lambda_i \) nonzero.

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

\[
\sum_{k=0}^{d} \lambda_k b^k = x, \quad \sum_{k=0}^{d} \lambda_k y^k = y,
\]

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

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

Implemented in most MILP solvers
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 \( \lambda_k \geq 0, \quad k = 0, 1, \ldots, d \) \((\lambda_0, \lambda_1, \ldots, \lambda_d)\) is SOS2

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

1. Select index \(k \in \{1, \ldots, d\}\) such that:
   \(\exists j_1 < k \text{ with } \lambda_{j_1} > 0\) and \(\exists j_2 > k \text{ with } \lambda_{j_2} > 0\)

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

1. MISOCPl Cuts
2. Global Optimization of Nonconvex MINLP
   - Challenges of Nonconvex MINLP
   - General Approach to Nonconvex MINLP
3. Piecewise Linear Approach to Nonconvex MINLP
   - Piecewise Linear Approach to Univariate Nonconvex MINLP
   - Piecewise Linear Approach to Multivariate Nonconvex MINLP
   - Beyond Piecewise Linear Functions
   - A Branch-and-Refine Algorithm
4. Summary and Student Discussion
SOS-2 generalizes to multiple dimensions [Beale and Tomlin, 1970]

- Multivariate $g : \mathbb{R}^d \rightarrow \mathbb{R}$
- Piecewise linear approx. of $g(x)$
- Choose breakpoints $b^k$, $k = 1, \ldots, q$
- Partition $\bigotimes_{i=1}^d [l_i, u_i]$ 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 $d + 1$ $\lambda_k$ 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. \( \lambda_{ij} \) 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. \( \lambda_{ij} \) weight of vertex \((i, j)\)
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. \( \lambda_{ij} \) weight of vertex \((i, j)\)

\[
\begin{align*}
v &= \sum_{i=1}^{k} \lambda_{ij} v_i, \\
w &= \sum_{j=1}^{l} \lambda_{ij} w_j, \\
u &= \sum_{i=1}^{k} \sum_{j=1}^{l} \lambda_{ij} u_{ij}, \\
1 &= \sum_{i=1}^{k} \sum_{j=1}^{l} \lambda_{ij} \text{ is SOS3} . . .
\end{align*}
\]
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

\[ \iff \exists \ \text{triangle} \ \Delta : \{(i,j) : \lambda_{ij} > 0\} \subset \Delta \]

i.e. nonzeros in single triangle \( \Delta \)
Branching on SOS3 when $\lambda$ violates set condition

- compute centers:
  \[ \hat{v} = \sum \lambda_{ij} v_i \quad \& \quad \hat{w} = \sum \lambda_{ij} w_i \]
- find $s, t$ such that
  \[ v_s \leq \hat{v} < v_{s+1} \quad \& \quad w_t \leq \hat{w} < w_{t+1} \]
- branch on $v$ or $w$
Branching on SOS3 when $\lambda$ violates set condition

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

**Vertical branching:**

\[ \sum_{L} \lambda_{ij} = 1 \quad \text{&} \quad \sum_{R} \lambda_{ij} = 1 \]
Piecewise Linear Approximations for Multivariate Functions

Branching on SOS3 when $\lambda$ 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 $w$

horizontal branching:

$$\sum_{T} \lambda_{ij} = 1$$
$$\sum_{B} \lambda_{ij} = 1$$
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 $\lambda_i$

- 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 $\lambda_k$

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

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

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.

- \( w_j = x_j \quad j = 1, \ldots, \)
- \( w_5 = w_2^2 \)
- \( w_6 = w_2w_3 \)
- \( w_7 = \cos(w_4) \)
- \( w_8 = w_6w_7 \)
- \( g = aw_5 + bw_8 - w_1 \)

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:
- \( \nu_i^2 \)
- \( \cos(\zeta_{ij}) \)
- \( \sin(w_{j1}) \), where \( w_{j1} = \zeta_{ij} + \theta_i - \theta_j \)
- 5 bilinear terms like \( \nu_i \nu_j \)

\[ \Rightarrow \text{need only } 5p^2 + 3p \text{ SOS variables, } \lambda \ldots \text{ much smaller } p^8 \]
Consider MINLP in format

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

... 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

\[
(P) \left\{ \begin{array}{l}
\min_x g_0(x), \\
\text{s.t. } g_i(x) \leq 0, \ i = 1, \ldots, m, \\
x \in X, \ x_i \in \mathbb{Z}^p
\end{array} \right. \]

Introduce variables \( w \), write MINLP \((P)\) equivalently as

\[
(D) \left\{ \begin{array}{l}
\text{minimize } \ w_0, K_0 \\
\text{subject to } \ w_{ij} = x_j \\
\ w_{i,n+j} = g_{ij}(w_{i,j_1}, w_{i,j_2}) \ \\
\ w_{i,n_i} \leq 0 \\
x \in X, \ x_i \in \mathbb{Z} \\
\end{array} \right. \forall i, \forall j
\]

\[
\ldots \text{equivalent to MINLP } (P) \ldots \text{ related to automatic differentiation}
\]

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

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

Get factorable form:

\[
\begin{align*}
  w_{11} &= \nu_i, \\
  w_{12} &= \nu_i^2, \\
  w_{12} &= w_{11}, \\
  w_{33} &= \cos(w_{31} + \theta_i - \theta_j), \\
  w_{34} &= w_{33} w_{22}, \\
  w_{35} &= y_{ij} w_{32} + g_{ij} \\
  w_{36} &= w_{35} w_{12}
\end{align*}
\]

BARON & Couenne solvers use factorable format.
Outline

1. MISOCPP Cuts

2. Global Optimization of Nonconvex MINLP
   - Challenges of Nonconvex MINLP
   - General Approach to Nonconvex MINLP

3. Piecewise Linear Approach to Nonconvex MINLP
   - Piecewise Linear Approach to Univariate Nonconvex MINLP
   - Piecewise Linear Approach to Multivariate Nonconvex MINLP
   - Beyond Piecewise Linear Functions
   - A Branch-and-Refine Algorithm

4. Summary and Student Discussion
SOS Approximations Become Infeasible

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

\ldots observed infeasible SOS on some power-grid examples!
SOS Approximations Become Infeasible

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

... observed infeasible SOS on some power-grid examples!
SOS Approximations Become 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:

\[
\sum_{k \in I} \lambda_k (g(w_k) - L_k) \leq w_g \leq \sum_{k \in I} \lambda_k (g(w_k) + U_k)
\]
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. $g(w) = w^2$:

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

See Emilie’s thesis for other functions ...
Piecewise Polyhedral Envelopes for $g = xy$

**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):

$$w_{xy} = \sum_{(i,j) \in I} \lambda_{ij} x_i y_j$$
Proposition: \((E)\) is an outer approximation of \((D)\) and hence \((P)\).

\[
\begin{align*}
\text{minimize} & \quad w_0, K_0 \\
\text{subject to} & \quad w_{ij} = x_j, \\
& \quad x_j = \sum_{k \in I_j} \lambda_{jk} x_{jk}, \quad 1 = \sum_{k \in I_j} \lambda_{jk} \\
& \quad w_{i,n+j} \geq \sum_{k \in I_{ij}} \lambda_{ij}^k \left( g_{ij}(w_{i,j1}^k, w_{i,j2}^k) - L_{ijk} \right) \\
& \quad w_{i,n+j} \leq \sum_{k \in I_{ij}} \lambda_{ij}^k \left( g_{ij}(w_{i,j1}^k, w_{i,j2}^k) + U_{ijk} \right) \\
& \quad w_{i,s+K_i} = 0 \\
& \quad x \in X, \quad x \in \mathbb{Z}^p, \quad \text{and} \quad w \in W,
\end{align*}
\]

where \(W\) deduced from \(x\) bounds; and blue part replaces \(w_{i,n+j} = g_{ij}(w_{i,j1}^k, w_{i,j2}^k)\)
Piecewise Envelope Problem: Illustration

SOS Outer Approximation

Convex Hull
Outline

1. MISOCP Cuts

2. Global Optimization of Nonconvex MINLP
   - Challenges of Nonconvex MINLP
   - General Approach to Nonconvex MINLP

3. Piecewise Linear Approach to Nonconvex MINLP
   - Piecewise Linear Approach to Univariate Nonconvex MINLP
   - Piecewise Linear Approach to Multivariate Nonconvex MINLP
   - Beyond Piecewise Linear Functions
   - A Branch-and-Refine Algorithm

4. Summary and Student Discussion
Branch-and-Refine: Outline

Classical Branch-and-Bound:
Solve envelope problem \((E)\) branch on SOS-condition or \(x_I \in \mathbb{Z}^p\)
\[ \Rightarrow \text{large discretization error or large number of } \lambda_k \text{ variables} \]

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

1D SOS

2D SOS
Also solve $\text{NLP}(X_k)$:

$$
\begin{cases}
  z_{\text{NLP}_k} := \min_x g_0(x) \\
  \text{subject to } g_i(x) = 0, \ i = 1, \ldots, m \\
  x \in X_k,
\end{cases}
$$

...upper bound on node $(X_k)$.

**Fathoming Rules:**

1. infeasible LP relaxation
2. $\text{NLP}(X_k)$ solution same as $\text{LP}(X_k)$ relaxation
3. LP relaxation dominated by incumbent
Branch-and-Refine: Algorithm

set \( U = \infty \), \( k = 1 \) & put \( \text{LP}(X_k) \) on stack

while stack is not empty

solve \( \text{LP}(X_k) \) ... solution \( x^k \)

if \( \text{LP}(X_k) \) infeasible or \( z_{\text{LP}_k} \geq U - \epsilon \) then

fathom node (case 1. or 3.)

else

solve \( \text{NLP}(X_k) \) ... solution \( \hat{x}^k \)

if \( z_{\text{NLP}_k} < U - \epsilon \) & \( \hat{x}^k \) integer then

update \( U := z_{\text{NLP}_k} \) & incumbent \( x^* := \hat{x} \)

if \( |z_{\text{NLP}_k} - z_{\text{LP}_k}| \leq \epsilon \) then

fathom node (case 2.)

else

branch creating two new LPs

Theorem: If \( x \in X \) is bounded \( \Rightarrow \) get \( \epsilon \)-optimal solution.
## Test Problems (Generic)

<table>
<thead>
<tr>
<th>prob</th>
<th>#var</th>
<th>#cons</th>
<th>#var OA</th>
<th>#cons OA</th>
<th>#sets λ</th>
<th>#disc</th>
</tr>
</thead>
<tbody>
<tr>
<td>pb0</td>
<td>4</td>
<td>2</td>
<td>44</td>
<td>32</td>
<td>6</td>
<td>1</td>
</tr>
<tr>
<td>pb1</td>
<td>4</td>
<td>2</td>
<td>44</td>
<td>32</td>
<td>6</td>
<td>1</td>
</tr>
<tr>
<td>pb2</td>
<td>6</td>
<td>2</td>
<td>41</td>
<td>30</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>pb3</td>
<td>6</td>
<td>2</td>
<td>41</td>
<td>30</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>pb4</td>
<td>12</td>
<td>4</td>
<td>97</td>
<td>71</td>
<td>11</td>
<td>2</td>
</tr>
<tr>
<td>pb5</td>
<td>12</td>
<td>4</td>
<td>97</td>
<td>71</td>
<td>11</td>
<td>2</td>
</tr>
<tr>
<td>pb6</td>
<td>12</td>
<td>4</td>
<td>143</td>
<td>97</td>
<td>19</td>
<td>3</td>
</tr>
<tr>
<td>pb7</td>
<td>12</td>
<td>4</td>
<td>143</td>
<td>97</td>
<td>19</td>
<td>3</td>
</tr>
<tr>
<td>pb8</td>
<td>12</td>
<td>4</td>
<td>119</td>
<td>77</td>
<td>14</td>
<td>2</td>
</tr>
<tr>
<td>pb9</td>
<td>12</td>
<td>4</td>
<td>119</td>
<td>77</td>
<td>14</td>
<td>2</td>
</tr>
<tr>
<td>pb10</td>
<td>10</td>
<td>4</td>
<td>111</td>
<td>72</td>
<td>13</td>
<td>2</td>
</tr>
<tr>
<td>pb11</td>
<td>10</td>
<td>4</td>
<td>111</td>
<td>72</td>
<td>13</td>
<td>2</td>
</tr>
<tr>
<td>pb12</td>
<td>24</td>
<td>8</td>
<td>275</td>
<td>187</td>
<td>40</td>
<td>6</td>
</tr>
<tr>
<td>pb13</td>
<td>24</td>
<td>8</td>
<td>275</td>
<td>187</td>
<td>40</td>
<td>6</td>
</tr>
</tbody>
</table>
## Test Problems (Tertiary Voltage Control)

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

... moderately sized problems

Complexity of nonconvex MINLPs depends on

\[ \# \text{ terms in computational graph} \sim \#\text{sets } \lambda \]
Do We Need Global Solvers?

<table>
<thead>
<tr>
<th>Solver</th>
<th># Problems Solved</th>
<th># Global Solutions</th>
</tr>
</thead>
<tbody>
<tr>
<td>BnR</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>Filter</td>
<td>12</td>
<td>8</td>
</tr>
<tr>
<td>IPOPT</td>
<td>17</td>
<td>14</td>
</tr>
<tr>
<td>KNITRO</td>
<td>17</td>
<td>13</td>
</tr>
</tbody>
</table>

**Comparison with MINLP solvers**

<table>
<thead>
<tr>
<th>Solver</th>
<th># Problems Solved</th>
<th># Global Solutions</th>
</tr>
</thead>
<tbody>
<tr>
<td>BnR</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>BONMIN</td>
<td>15</td>
<td>11</td>
</tr>
<tr>
<td>MINLPBB</td>
<td>11</td>
<td>9</td>
</tr>
</tbody>
</table>
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)

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

<table>
<thead>
<tr>
<th>prob</th>
<th>basic</th>
<th>+presolve</th>
<th>+var-select</th>
<th>+node-select</th>
</tr>
</thead>
<tbody>
<tr>
<td>TVC1</td>
<td>108861</td>
<td>40446</td>
<td>7756</td>
<td>8031</td>
</tr>
<tr>
<td>TVC2</td>
<td>fail</td>
<td>72270</td>
<td>5792</td>
<td>5547</td>
</tr>
<tr>
<td>TVC3</td>
<td>62045</td>
<td>861</td>
<td>627</td>
<td>627</td>
</tr>
<tr>
<td>TVC4</td>
<td>fail</td>
<td>38792</td>
<td>1396</td>
<td>1582</td>
</tr>
<tr>
<td>TVC5</td>
<td>fail</td>
<td>7369</td>
<td>5619</td>
<td>4338</td>
</tr>
<tr>
<td>TVC6</td>
<td>fail</td>
<td>12131</td>
<td>6096</td>
<td>5503</td>
</tr>
</tbody>
</table>
Outline

1. MISOCPP Cuts

2. Global Optimization of Nonconvex MINLP
   - Challenges of Nonconvex MINLP
   - General Approach to Nonconvex MINLP

3. Piecewise Linear Approach to Nonconvex MINLP
   - Piecewise Linear Approach to Univariate Nonconvex MINLP
   - Piecewise Linear Approach to Multivariate Nonconvex MINLP
   - Beyond Piecewise Linear Functions
   - A Branch-and-Refine Algorithm

4. Summary and Student Discussion
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


