Generalized Surrogate Duality in Mixed-Integer Nonlinear Programming

Benjamin Müller², Gonzalo Muñoz¹, Maxime Gasse¹, Ambros Gleixner², Andrea Lodi¹, Felipe Serrano²
FIFTY, Berlin - September 27, 2019

¹Polytechnique Montréal, Canada
²Zuse Institute Berlin, Germany
Nonconvex MINLPs
Mixed-Integer Nonlinear Programs (MINLPs)

\[
\begin{align*}
\min & \quad c^T x \\
\text{s.t.} & \quad g_k(x) \leq 0 \quad \forall k \in [m] \\
& \quad Ax \leq b \\
& \quad x_i \in \mathbb{Z} \quad \forall i \in \mathcal{I} \subseteq [n] \\
& \quad x_i \in [\ell_i, u_i] \quad \forall i \in [n]
\end{align*}
\]

The functions \( g_k : [\ell, u] \to \mathbb{R} \) can be

- **convex**
- **nonconvex**
MINLP is “The mother of all deterministic optimization problems”

(Jon Lee, 2008)
Solving MINLPs

MINLP is “The mother of all deterministic optimization problems”

(Jon Lee, 2008)

- Source of difficulty: Nonconvex nonlinearities
- Main challenges:
  - Convexification of nonconvex nonlinearities
  - Reduction of convexification gap (spatial branch-and-bound)
  - Numerical robustness
  - Diversity of problem class
Solving MINLPs

Cuts

Underestimators

Spatial branching

Reformulation

Bound tightening

Primal heuristics
Performance of solvers depends on tight relaxations.
Importance of good relaxations

Performance of solvers depends on tight relaxations.

For efficiency, typically they are convex (most of the time linear).
Performance of solvers depends on tight relaxations.

For efficiency, typically they are convex (most of the time linear).

Here: we consider a nonconvex relaxation.
Surrogate Duality
Given $\lambda \in \mathbb{R}^m_+$ we define the surrogate relaxation $S(\lambda)$ as

$$\min c^T x$$

subject to

$$\sum_{k \in [m]} \lambda_k g_k(x) \leq 0$$

$$Ax \leq b$$

$$x_i \in \mathbb{Z} \quad \forall i \in \mathcal{I} \subseteq [n]$$

$$x_i \in [l_i, u_i] \quad \forall i \in [n]$$

We denote its feasible region $S_\lambda$. 
Surrogate relaxation - Observations

Dates back to Glover (1965)
Dates back to Glover (1965)

- $S_\lambda$ is a valid relaxation for any $\lambda \geq 0$
Surrogate relaxation - Observations

Dates back to Glover (1965)

- $S_\lambda$ is a valid relaxation for any $\lambda \geq 0$

- The surrogate relaxation is still a (potentially hard) MINLP
Surrogate relaxation - Observations

Dates back to Glover (1965)

- $S_\lambda$ is a valid relaxation for any $\lambda \geq 0$

- The surrogate relaxation is still a (potentially hard) MINLP

- But! A single nonconvex constraint $\Rightarrow$ simpler to optimize over
Surrogate relaxation - Example

\[
\begin{align*}
\min & \quad -y \\
\text{s.t.} & \quad g_1(x, y) := 2xy + x^2 - y^2 - x \leq 0 \\
& \quad g_2(x, y) := -xy - 0.3x^2 - 0.2y^2 - 0.5x + 1.5y \leq 0 \\
& \quad x, y \in [0, 1]
\end{align*}
\]
Surrogate relaxation - Example

\[
\begin{align*}
\min & \quad -y \\
\text{s.t.} & \quad g_1(x, y) := 2xy + x^2 - y^2 - x \leq 0 \\
& \quad g_2(x, y) := -xy - 0.3x^2 - 0.2y^2 - 0.5x + 1.5y \leq 0 \\
& \quad x, y \in [0, 1]
\end{align*}
\]

Surrogate relaxation:

\[
\begin{align*}
\min & \quad -y \\
\text{s.t.} & \quad \lambda g_1(x, y) + (1 - \lambda)g_2(x, y) \leq 0 \\
& \quad x, y \in [0, 1]
\end{align*}
\]
Surrogate relaxation - Example

\[ S_\lambda := \{(x, y) \in [0, 1]^2 \mid \lambda g_1(x, y) + (1 - \lambda)g_2(x, y) \leq 0\} \]
Surrogate relaxation - Example

\[ S_\lambda := \{(x, y) \in [0, 1]^2 \mid \lambda g_1(x, y) + (1 - \lambda)g_2(x, y) \leq 0\} \]

\[ \lambda = \frac{3}{4} \]
Surrogate relaxation - Example

\[ S_\lambda := \{(x, y) \in [0, 1]^2 \mid \lambda g_1(x, y) + (1 - \lambda) g_2(x, y) \leq 0\} \]

\[
\lambda = \frac{3}{4} \quad \text{and} \quad \lambda = \frac{2}{3}
\]
Surrogate relaxation - Example

\[ S_\lambda := \{(x, y) \in [0, 1]^2 \mid \lambda g_1(x, y) + (1 - \lambda)g_2(x, y) \leq 0\} \]

\( \lambda = \frac{3}{4} \quad \lambda = \frac{2}{3} \quad \lambda = \frac{1}{2} \)
Surrogate relaxation - Example

\[ S_\lambda := \{(x, y) \in [0, 1]^2 \mid \lambda g_1(x, y) + (1 - \lambda)g_2(x, y) \leq 0\} \]

\[ \lambda = \frac{3}{4} \quad \lambda = \frac{2}{3} \quad \lambda = \frac{1}{2} \quad \lambda = \frac{1}{4} \]
Surrogate relaxation - Example

\[ S_\lambda := \{(x, y) \in [0, 1]^2 \mid \lambda g_1(x, y) + (1 - \lambda)g_2(x, y) \leq 0\} \]

Observations

- \( S_\lambda \) can be disconnected, nonconvex, or convex
Surrogate relaxation - Example

\[ S_\lambda := \{(x, y) \in [0, 1]^2 \mid \lambda g_1(x, y) + (1 - \lambda)g_2(x, y) \leq 0\} \]

\[ \lambda = \frac{3}{4} \quad \lambda = \frac{2}{3} \quad \lambda = \frac{1}{2} \quad \lambda = \frac{1}{4} \]

Observations

- \( S_\lambda \) can be disconnected, nonconvex, or convex
- \( \min_{x,y} \{-y \mid (x, y) \in S_\lambda\} \) is a valid dual bound for every \( \lambda \)
Surrogate relaxation - Example

\[ S_\lambda := \{(x, y) \in [0, 1]^2 \mid \lambda g_1(x, y) + (1 - \lambda)g_2(x, y) \leq 0\} \]

\[ \lambda = \frac{3}{4} \quad \lambda = \frac{2}{3} \quad \lambda = \frac{1}{2} \quad \lambda = \frac{1}{4} \]

Observations

- \( S_\lambda \) can be disconnected, nonconvex, or convex
- \( \min_{x,y} \{-y \mid (x, y) \in S_\lambda\} \) is a valid dual bound for every \( \lambda \)
- \( \lambda = \frac{1}{2} \) results in a good relaxation \( \implies \) what is the best \( \lambda \)?
Surrogate duality

Finding best $\lambda$ → surrogate dual:

$$\sup_{\lambda} \{ S(\lambda) := \min_{x} \{ c^T x \mid x \in S_\lambda \} \}$$
Surrogate duality

Finding best $\lambda \rightarrow$ surrogate dual:

$$\sup_{\lambda} \{S(\lambda) := \min_{x} \{c^T x \mid x \in S_\lambda\}\}$$

Observations

- Complex bilevel optimization problem
Finding best $\lambda \underset{\rightarrow}{\rightarrow}$ surrogate dual:

$$\sup_{\lambda} \{ S(\lambda) := \min_{x} \{ c^T x \mid x \in S_{\lambda} \} \}$$

Observations

- Complex bilevel optimization problem
- $S(\lambda)$ is quasi-concave, but can be \textit{discontinuous}
Surrogate duality

Finding best $\lambda \rightarrow$ surrogate dual:

$$\sup_{\lambda} \{ S(\lambda) := \min_{x} \{ c^T x \mid x \in S_{\lambda} \} \}$$

Observations

- Complex bilevel optimization problem
- $S(\lambda)$ is quasi-concave, but can be *discontinuous*

- $S(\lambda)$ is just lower semi-continuous $\rightarrow$ sup is not max
Solving the surrogate dual via Benders
A Benders algorithm

Goal: solve $\sup_{\lambda \geq 0} S(\lambda)$
A Benders algorithm

Goal: solve $\sup_{\lambda \geq 0} S(\lambda)$

Intuition
Let $x^*$ be a solution of $S(\lambda)$. Can we find $\lambda^*$ such that $x^*$ is infeasible?
A Benders algorithm

Goal: solve $\sup_{\lambda \geq 0} S(\lambda)$

Intuition

Let $x^*$ be a solution of $S(\lambda)$. Can we find $\lambda^*$ such that $x^*$ is infeasible?

Proposed independently by Banerjee (1971), Karwan (1976) and Dyer (1980)
A Benders algorithm

The following LP finds an aggregation that makes every $\bar{x} \in \mathcal{P}$ infeasible.

$$\begin{align*}
\max_{\lambda} & \quad \Psi \\
\text{s.t.} & \quad \sum_{j \in [m]} \lambda_j g_j(\bar{x}) \geq \Psi \quad \forall \bar{x} \in \mathcal{P} \\
& \quad \lambda_j \geq 0 \quad \forall j \in [m] \\
& \quad \sum_{j \in [m]} \lambda_j = 1
\end{align*}$$
A Benders algorithm
A Benders algorithm

\[ \lambda^* = (1, 0) \]
A Benders algorithm

\[ \lambda^* = (0, 1) \]
A Benders algorithm

\[ \lambda^* = (0.473, 0.526) \]
A Benders algorithm

\[ \lambda^* = (0.539, 0.460) \]
A Benders algorithm

\[ \lambda^* = (0.562, 0.437) \]
A Benders algorithm

\[ \lambda^* = (0.565, 0.434) \]
A Benders algorithm

\[ \lambda^* = (0.563, 0.436) \]
Guarantees of the algorithm

Theorem (Karwan 1976)

Denote by \((\lambda^t, \Psi^t)\) the sequence of solutions obtained from the master problem of the Benders algorithm. The algorithm either

- **Terminates in** \(T\) **steps, in which case**

  \[
  \max_{1 \leq t \leq T} S(\lambda^t) = \sup_{\lambda} S(\lambda)
  \]

- **Otherwise,**

  \[
  \sup_{t \geq 1} S(\lambda^t) = \sup_{\lambda} S(\lambda)
  \]
Practical improvements

**Bottleneck:** Solving $S(\lambda)$
Practical improvements

Bottleneck: Solving $S(\lambda)$

Improvements

- Improved LP relaxation from SCIP’s root node processing
Bottleneck: Solving $S(\lambda)$

Improvements

- Improved LP relaxation from SCIP’s root node processing
- Early stopping when

best primal solution of $S(\lambda) \leq$ best dual bound in Benders
Practical improvements

Bottleneck: Solving $S(\lambda)$

Improvements

- Improved LP relaxation from SCIP’s root node processing
- Early stopping when
  
  \[ \text{best primal solution of } S(\lambda) \leq \text{best dual bound in Benders} \]

- Dual objective cutoff and target dual improvement
Table 1: Gap closed by Benders algorithm on SCIP’s internal MINLP test set containing 143 instances.
Generalized Surrogate Duality
Main idea

In problems with many constraints, consider $K$ aggregations

$$\sum_{j \in [m]} \lambda_j^k g_j(x) \leq 0, \quad k \in \{1, \ldots, K\}$$

Denote $S^K(\lambda)$ the resulting relaxation.
Main idea

In problems with many constraints, consider $K$ aggregations

$$\sum_{j \in [m]} \lambda_j^k g_j(x) \leq 0, \quad k \in \{1, \ldots, K\}$$

Denote $S^K(\lambda)$ the resulting relaxation.

Main idea

In problems with many constraints, consider $K$ aggregations

$$\sum_{j \in [m]} \lambda_j^k g_j(x) \leq 0, \quad k \in \{1, \ldots, K\}$$

Denote $S^K(\lambda)$ the resulting relaxation.


Moreover, Karwan and Rardin (1980) argue that the loss of “desirable properties” of the generalization impairs the guarantees of search procedures.
Generalized Surrogate Duality - Issues

Issues

- Surrogate dual is no longer quasiconcave
- Surrogate relaxations become computationally harder
Generalized Surrogate Duality - Issues

Issues

- Surrogate dual is no longer quasiconcave
- Surrogate relaxations become computationally harder

Potential advantages

- Relaxation quality can be much better
- Benders algorithm could be adapted
Figure 1: Dual bounds for $K = 1, 2, 3$ for 600 iterations
Adapting Benders

Master problem

$S^K(\lambda)$

$\lambda^*$

making $x^*$ infeasible

$x^*$

optimal
Adapting Benders

Master problem

$S^K(\lambda)$

$\lambda^*$

making $x^*$ infeasible

$\ast$ making $x^*$ optimal
$K$ aggregations are encoded by

$$\lambda = (\lambda_1, \lambda_2, \ldots, \lambda^K) \in \mathbb{R}_+^{Km}$$
Adapting Benders

$K$ aggregations are encoded by

$$\lambda = (\lambda^1, \lambda^2, \ldots, \lambda^K) \in \mathbb{R}^{Km}_+$$

The master problem becomes

$$\begin{align*}
\text{max } & \Psi, \\
\text{s.t. } & \left( \sum_{j \in [m]} \lambda^1_j g_j(\bar{x}) \geq \Psi \right) \lor \ldots \lor \left( \sum_{j \in [m]} \lambda^K_j g_j(\bar{x}) \geq \Psi \right) \quad \forall \bar{x} \in \mathcal{P}, \\
& \|\lambda^k\|_1 \leq 1, \lambda^k \geq 0 \quad \forall k \in \{1, \ldots, K\},
\end{align*}$$

which can be cast as a MIP $\rightsquigarrow$ potential bottleneck.
Convergence result

**Theorem**

Denote by \((\lambda^t, \Psi^t)\) the sequence of solutions obtained from the master problem of the generalized Benders algorithm. The algorithm either

- **Terminates in** \(T\) steps, in which case

  \[
  \max_{1 \leq t \leq T} S^K(\lambda^t) = \sup_{\lambda \geq 0} S^K(\lambda)
  \]

- **Otherwise,**

  \[
  \sup_{t \geq 1} S^K(\lambda^t) = \sup_{\lambda \geq 0} S^K(\lambda)
  \]
Computational enhancements

- Improved LP relaxation from SCIP’s root node processing
- Early stopping of $S^K(\lambda)$ when
  
  best primal solution of $S(\lambda) \leq$ best dual bound in Benders

- Dual-objective cutoff and target dual improvement
- Improved LP relaxation from SCIP’s root node processing
- Early stopping of $S^K(\lambda)$ when
  
  \[
  \text{best primal solution of } S(\lambda) \leq \text{best dual bound in Benders}
  \]
- Dual-objective cutoff and target dual improvement
- Early stopping of the master problem
Computational enhancements

- Improved LP relaxation from SCIP’s root node processing
- Early stopping of $S^K(\lambda)$ when
  best primal solution of $S(\lambda) \leq$ best dual bound in Benders
- Dual-objective cutoff and target dual improvement
- Early stopping of the master problem
- Support stabilization
Computational enhancements

- Improved LP relaxation from SCIP’s root node processing
- Early stopping of $S^K(\lambda)$ when
  
  best primal solution of $S(\lambda) \leq$ best dual bound in Benders

- Dual objective cutoff and target dual improvement
- Early stopping of the master problem
- Support stabilization
- Box stabilization
Computational enhancements

- **Improved LP relaxation** from SCIP’s root node processing
- **Early stopping of** $S^K(\lambda)$ **when**
  
  best primal solution of $S(\lambda) \leq$ best dual bound in Benders

- **Dual-objective cutoff** and **target** dual improvement
- **Early stopping of the** master problem
- **Support** stabilization
- **Box** stabilization
- **Symmetry breaking** over multiplier vectors
Computational Experiments
<table>
<thead>
<tr>
<th>group</th>
<th># instances</th>
<th>$K = 1$</th>
<th>$K = 2$</th>
<th>$K = 3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALL(^1)</td>
<td>633</td>
<td>18.4%</td>
<td>21.4%</td>
<td>23.4%</td>
</tr>
<tr>
<td>$m \geq 10$</td>
<td>528</td>
<td>14.6%</td>
<td>16.9%</td>
<td>18.4%</td>
</tr>
<tr>
<td>$m \geq 20$</td>
<td>391</td>
<td>10.7%</td>
<td>12.3%</td>
<td>13.5%</td>
</tr>
<tr>
<td>$m \geq 50$</td>
<td>229</td>
<td>7.1%</td>
<td>7.9%</td>
<td>8.5%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>AFFECTED</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALL</td>
</tr>
<tr>
<td>$m \geq 10$</td>
</tr>
<tr>
<td>$m \geq 20$</td>
</tr>
<tr>
<td>$m \geq 50$</td>
</tr>
</tbody>
</table>

**Table 2:** Gap closed w.r.t the MIP relaxation $S(0)$.

\(^1\)instances with at least 4 nonlinear constraints; not solved in the root node.
Improving best known dual bounds on MINLPLib

After 3h default SCIP, set target dual bound to close 20% of the gap. Average optimality gap over 209 left instances reduces from 284.3% to 142.8%.

In particular:

<table>
<thead>
<tr>
<th>instance</th>
<th>PB</th>
<th>DB</th>
<th>MINLPLib</th>
<th>new DB</th>
</tr>
</thead>
<tbody>
<tr>
<td>polygon25</td>
<td>-0.78</td>
<td>-5.80</td>
<td>-3.94</td>
<td></td>
</tr>
<tr>
<td>polygon50</td>
<td>-0.78</td>
<td>-15.27</td>
<td>-8.72</td>
<td></td>
</tr>
<tr>
<td>polygon75</td>
<td>-0.78</td>
<td>-24.87</td>
<td>-13.55</td>
<td></td>
</tr>
<tr>
<td>polygon100</td>
<td>-0.78</td>
<td>-34.00</td>
<td>-19.03</td>
<td></td>
</tr>
<tr>
<td>sfacloc1_4_80</td>
<td>7.88</td>
<td>0.16</td>
<td>0.41</td>
<td></td>
</tr>
<tr>
<td>sfacloc1_4_90</td>
<td>10.46</td>
<td>0.48</td>
<td>1.18</td>
<td></td>
</tr>
<tr>
<td>sfacloc1_4_95</td>
<td>11.18</td>
<td>0.79</td>
<td>2.40</td>
<td></td>
</tr>
</tbody>
</table>
Take-away messages

- Classical and Generalized Surrogate dual can provide strong bounds
- A Benders algorithm can be used to find the best multipliers with convergence guarantees
- Computational enhancements can greatly help for achieving practicality
- Still too heavy for direct inclusion in branch-and-bound, but working towards it!
Thank you!

#surrogaterevival