LaGO

a Branch and Cut framework for nonconvex MINLPs

Ivo Nowak and Stefan Vigerske

Humboldt-University Berlin, Department of Mathematics

EURO XXI, July 5, 2006
21st European Conference on Operational Research, Reykjavik
Lagrangian Global Optimizer

General purpose solver for sparse, block-separable, nonconvex MINLPs

History:

2000 Development started by Ivo Nowak as a solver for nonconvex MIQQPs based on Lagrangian decomposition and semidefinite relaxation

2001-2004 Project funded by German Science Foundation: extension to MINLP solver

- Branch and Cut for MIQQPs
- heuristic Branch and Cut for nonconvex MINLPs
- start of Branch Cut and Price algorithm for MINLPs

Webpage: http://www.math.hu-berlin.de/~eopt/LaGO

Book: Ivo Nowak, Relaxation and Decomposition Methods for Mixed Integer Nonlinear Programming, Birkhäuser 2005
Overview

Preprocessing

Branch and Cut algorithm

Cutting planes

Boxreduction

Numerical Results

Further developments
We consider problems of the form

\[
\begin{align*}
\text{minimize} & \quad \mathbf{c}^T \mathbf{x} \\
\text{such that} & \quad h_i (\mathbf{x}) \leq 0, \quad i \in I, \\
& \quad h_j (\mathbf{x}) = 0, \quad j \in E, \\
& \quad x_k \in \{0, 1\}, \quad k \in B, \\
& \quad \mathbf{x} \in [\underline{x}, \bar{x}] \\
& \quad -\infty < x_i \leq \bar{x}^i < \infty, \quad i \in \{1, \ldots, n\}, \\
& \quad h \in C^2 ([\underline{x}, \bar{x}], \mathbb{R}^{|I|+|E|}), \\
& \quad \mathbf{c} \in \mathbb{R}^n
\end{align*}
\]

LaGO interfaces problems via GAMS
Overview

Preprocessing

Branch and Cut algorithm

Cutting planes

Boxreduction

Numerical Results

Further developments
Preprocessing

- Investigation of **problem structure** (sparsity, block-separability, quadratic functions, convexity).
- **Reduction of box** $[x, \bar{x}]$, determine bounding box for unbounded variables
- Initialization of **linear relaxation**:
  1. Nonquadratic nonconvex function $g$  
     $\Rightarrow$ quadratic (nonconvex) underestimator $q$
  2. Quadratic nonconvex function $q$  
     $\Rightarrow$ quadratic convex underestimator $\tilde{q}$
  3. Nonlinear convex function $\Rightarrow$ linearization
  4. Binary conditions are dropped.

Stefan Vigerske

LaGO - a Branch and Cut framework for nonconvex MINLPs
Let $g \in C^2([\bar{x}, \bar{x}], \mathbb{R})$ be nonquadratic. Consider a sample set $S \subseteq [\bar{x}, \bar{x}]$. We compute $q(x) = x^T A x + b^T x + c$ by minimization of

$$\sum_{x \in S} (g(x) - q(x)) + \delta_1 \sum_{x \in S_1} |\nabla (g - q)(x)|_1 + \delta_2 \sum_{x \in S_2} |\nabla^2 (g - q)(x)|_1$$

such that $q(x) \leq g(x)$ for all $x \in S$, where $S_2 \subseteq S_1 \subseteq S$ and $\delta_1, \delta_2 \geq 0$.

- Can be formulated as a linear program.
- Sparsity of $A$ and $b$ determined by $g(x)$. 

Nonconvex quadratic underestimator
Convex quadratic underestimator

Let \( q(x) = x^T Ax + b^T x + c \) be a quadratic nonconvex function.

A convex \( \alpha \)-underestimator (Adjiman and Floudas 1997) of \( q(x) \) is

\[
\tilde{q}(x) = q(x) + \sum_{i=1}^{n} \alpha_i (x_i - \bar{x}_i)(x_i - \bar{x}_i)
\]

where

\[
\alpha_i = -\lambda_1 (\text{Diag}(\bar{x} - x) A \text{Diag}(\bar{x} - x)) (\bar{x}_i - x_i)^{-2}.
\]
Overview

Preprocessing

**Branch and Cut algorithm**

Cutting planes

Boxreduction

Numerical Results

Further developments
Main Loop

Denote by $\hat{x}$ a solution of the linear relaxation.

1. Take node with lowest lower bound from Branch and Bound tree.
2. Upper bounds: Start local search (with fixed binary variables) from $\hat{x}$ (rounded) (GAMS/NLP-Solver or IPOPT)
Main Loop

Denote by $\hat{x}$ a solution of the linear relaxation.

1. Take node with lowest lower bound from Branch and Bound tree.

2. Upper bounds: Start local search (with fixed binary variables) from $\hat{x}$ (rounded) (GAMS/NLP-Solver or IPOPT)

3. Branch: select a variable $x_i$
   - whose binary condition is mostly violated by $\hat{x}$
   - or: where $g(x) \leq 0$ is mostly violated by $\hat{x}$, $\frac{\partial}{\partial x_i} g(\hat{x})$ is large, and the box of $x_i$ hasn’t been reduced very much so far
   - or: whose box is least reduced
Main Loop

Denote by \( \hat{x} \) a solution of the linear relaxation.

1. Take node with lowest lower bound from Branch and Bound tree.

2. Upper bounds: Start local search (with fixed binary variables) from \( \hat{x} \) (rounded) (GAMS/NLP-Solver or IPOPT)

3. Branch: select a variable \( x_i \)
   - whose binary condition is mostly violated by \( \hat{x} \)
   - or: where \( g(x) \leq 0 \) is mostly violated by \( \hat{x} \), \( \frac{\partial}{\partial x_i} g(\hat{x}) \) is large, and the box of \( x_i \) hasn’t been reduced very much so far
   - or: whose box is least reduced

4. Bound: for each child node
   4.1 Generate and update cuts
   4.2 Update the box
   4.3 Solve the linear relaxation (CPLEX or COIN/Clp)
   4.4 Put nodes into tree

5. Prune: Prune nodes which lower bound exceeds upper bound
Overview

Preprocessing

Branch and Cut algorithm

Cutting planes

Boxreduction

Numerical Results

Further developments
Linearization Cuts

reference point \( \hat{x} \), convex constraint \( g(x) \leq 0 \)

\[
g(\hat{x}) + \nabla g(\hat{x})(x - \hat{x}) \leq 0
\]

Linearizations of convexified functions

\[
g(x) := q(x) + \sum_{i=1}^{n} \alpha_i (x_i - x_i)(x_i - \bar{x}_i)
\]

can easily be updated after a reduction of the box \([\underline{x}, \bar{x}]\).
Mixed Integer Rounding Cuts

- Linear relaxation solved via COIN Open Solver Interface
- COIN Cut Generator Library provides several types of cuts to cut off a nonintegral solution of the relaxation

Mixed Integer Rounding Cut (Nemhauser, Wolsey 1988) principle:

\[ X := \{(x, y) \in \mathbb{Z} \times \mathbb{R}_+ | x - y \leq b\} \]

\[
x - \frac{1}{1 - (b - \lfloor b \rfloor)} y \leq \lfloor b \rfloor
\]

\[ \forall (x, y) \in X \]
Overview

Preprocessing

Branch and Cut algorithm

Cutting planes

Boxreduction

Numerical Results

Further developments
Boxreduction based on intervalarithmetic

- Consider a constraint

\[ g(x, y) := h(x) + y \leq 0, \]

i.e., \( y \leq -h(x) \), and let

\[ [h, \bar{h}] := -h([x, \bar{x}]). \]

- If \( \bar{h} \leq \bar{y} \), set

\[ \bar{y} := \bar{h} \]

and proceed with other constraints depending on \( y \).

- Does not rely on relaxations. Easy and fast to compute.

- Interval arithmetic provided by GAMS-interface and FILIB++. 

Stefan Vigerske

LaGO - a Branch and Cut framework for nonconvex MINLPs
Boxreduction based on linear relaxation

Consider a linear relaxation with constraints $Ax \leq b$.

Let $x^*$ be the best solution found so far.

$x_i := \min x_i$

\[ \text{s.t. } Ax \leq b \]
\[ c^T x \leq c^T x^* \]

$\bar{x}_i := \max x_i$

\[ \text{s.t. } Ax \leq b \]
\[ c^T x \leq c^T x^* \]
Overview

Preprocessing

Branch and Cut algorithm

Cutting planes

Boxreduction

Numerical Results

Further developments
GAMS MINLPLib and GlobalLib

- **at most 1000 variables** and **no integrality conditions** except for binary
  ⇒ 33 MIQQPs, 72 (nonquadratic) MINLPs, 166 QQPs
- **timelimit**: 1 hour
- **NLP subsolver**: CONOPT; **LP subsolver**: CPLEX 10.0

<table>
<thead>
<tr>
<th></th>
<th>MIQQPs</th>
<th>MINLPs</th>
</tr>
</thead>
<tbody>
<tr>
<td>number of models</td>
<td>33</td>
<td>72</td>
</tr>
<tr>
<td>best known optimal solution found</td>
<td>21</td>
<td>41</td>
</tr>
<tr>
<td>nonoptimal solution found</td>
<td>5</td>
<td>9</td>
</tr>
<tr>
<td>unsuccessful Branch &amp; Cut search</td>
<td>7</td>
<td>13</td>
</tr>
<tr>
<td>failure in preprocessing</td>
<td>0</td>
<td>9</td>
</tr>
</tbody>
</table>

Pentium IV 3.00 Ghz, 1 GB RAM, Linux 2.16.11
LaGO vs. BARON on MIQQP\textsc{}s

LaGO and BARON 7.5 on MIQQP\textsc{Ts} from MIN\textsc{L}PLib:

<table>
<thead>
<tr>
<th></th>
<th>Total</th>
<th>LaGO better</th>
<th>same</th>
<th>BARON better</th>
</tr>
</thead>
<tbody>
<tr>
<td>BARON fail, LaGO not</td>
<td>3</td>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LaGO faster</td>
<td>1</td>
<td></td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>both solvers the same</td>
<td>7</td>
<td></td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>BARON faster</td>
<td>15</td>
<td>2</td>
<td>12</td>
<td>1</td>
</tr>
<tr>
<td>LaGO fail, BARON not</td>
<td>2</td>
<td></td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>LaGO and BARON fail</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>33</td>
<td>5</td>
<td>23</td>
<td>5</td>
</tr>
</tbody>
</table>
LaGO vs. BARON on MINLPs

LaGO and BARON 7.5 on (nonquadratic) MINLPs from MINLPLib:

<table>
<thead>
<tr>
<th></th>
<th>Total</th>
<th>LaGO better</th>
<th>same</th>
<th>BARON better</th>
</tr>
</thead>
<tbody>
<tr>
<td>BARON fail, LaGO not</td>
<td>5</td>
<td>5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LaGO faster</td>
<td>10</td>
<td>1</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>both solvers the same</td>
<td>10</td>
<td>1</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>BARON faster</td>
<td>25</td>
<td>3</td>
<td>18</td>
<td>4</td>
</tr>
<tr>
<td>LaGO fail, BARON not</td>
<td>11</td>
<td></td>
<td></td>
<td>11</td>
</tr>
<tr>
<td>LaGO and BARON fail</td>
<td>11</td>
<td></td>
<td></td>
<td>11</td>
</tr>
<tr>
<td>Total</td>
<td>72</td>
<td>10</td>
<td>38</td>
<td>20</td>
</tr>
</tbody>
</table>

LaGO stops branching when all binary variables are fixed
LaGO vs. BARON on QQPs

running LaGO and BARON 7.5 on QQPs from GlobalLib:

<table>
<thead>
<tr>
<th></th>
<th>Total</th>
<th>LaGO better</th>
<th>same</th>
<th>BARON better</th>
</tr>
</thead>
<tbody>
<tr>
<td>BARON fail, LaGO not</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LaGO faster</td>
<td>11</td>
<td>1</td>
<td>9</td>
<td>1</td>
</tr>
<tr>
<td>both solvers the same</td>
<td>90</td>
<td>89</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>BARON faster</td>
<td>61</td>
<td>61</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LaGO fail, BARON not</td>
<td>3</td>
<td></td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>Total</td>
<td>166</td>
<td>2</td>
<td>159</td>
<td>5</td>
</tr>
</tbody>
</table>
Optimizing the design of a complex energy conversion plant

2001-2004: project funded by German Science Foundation

Institute for Energy Engineering (Technical University Berlin)
T. Ahadi-Oskui, F. Cziesla, G. Tsatsaronis

Institute for Mathematics (Humboldt University Berlin)
H. Alperin, I. Nowak, S. Vigerske

- Model: superstructure of a combined-cycle-based cogeneration plant
- Simultaneous structural and process variable optimization

picture: exhaust gas path through heat-recovery steam generator
Model of a complex energy conversion plant

- superstructure for electric power output of ≤ 400 MW and process steam production of ≤ 500 t/h
- degrees of freedom: 27 structural and 48 process variables
- constraints:
  - logic of the superstructure (connecting binary variables)
  - thermodynamic behavior (highly nonlinear), mass+energy balances
  - purchase equipment costs
- objective: total cost for cogeneration plant investment cost, operation and maintenance cost, taxes and insurances,…
- MINLP model: 1308 variables (44 binary) and 1659 constraints
- GAMS MINLPLib models super1, super2, super3, and super3t
Distributed genetic algorithm

- individual = set of decision variables
- fitness obtained by simulation of the superstructure

**HSC-GA**: hierarchical social competition algorithm:

- handling several populations in parallel
- organized by fitness of inhabitants
- individuals from lower population can move into subpopulation at higher level
- after evolving for some time, they migrate into higher population
Optimization of the superstructure

- HSC-GA and LaGO run for 24 hours
  - HSC-GA: \( \approx 20000 \) generations
  - LaGO: \( \approx 30000 \) Branch and Bound iterations

<table>
<thead>
<tr>
<th>demand</th>
<th>method</th>
<th>efficiency</th>
<th>cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>electric power: 300 MW</td>
<td>LaGO</td>
<td>56.7%</td>
<td>12674 Euro/h</td>
</tr>
<tr>
<td></td>
<td>HSC-GA</td>
<td>55.4%</td>
<td>12774 Euro/h</td>
</tr>
<tr>
<td>electric power: 290 MW</td>
<td>LaGO</td>
<td>68.5%</td>
<td>13424 Euro/h</td>
</tr>
<tr>
<td>process steam: 150 t/h</td>
<td>HSC-GA</td>
<td>67.7%</td>
<td>13399 Euro/h</td>
</tr>
<tr>
<td>electric power: 400 MW</td>
<td>LaGO</td>
<td>58.6%</td>
<td>16771 Euro/h</td>
</tr>
<tr>
<td></td>
<td>HSC-GA</td>
<td>58.7%</td>
<td>17229 Euro/h</td>
</tr>
</tbody>
</table>

Overview

Preprocessing

Branch and Cut algorithm

Cutting planes

Boxreduction

Numerical Results

Further developments
Mixed-integer linear relaxation

- MIP solver are fast and robust today.
- Replace linear relaxation by a mixed-integer linear relaxation.

Allows use of intervalgradient cuts (Boddy, Johnson 2003 for MIQQPs):

**Intervalgradient** of $g$:

$$[d, \bar{d}] := \nabla g ([\underline{x}, \bar{x}])$$

$$(\nabla g (x) \in [d, \bar{d}] \ \forall x \in [\underline{x}, \bar{x}])$$

**Intervalgradient cut** w.r.t. $\hat{x} \in [\underline{x}, \bar{x}]$:

$$g (\hat{x}) + \min_{d \in [d, \bar{d}]} d^T (x - \hat{x}) \leq 0$$
Intervalgradient Cuts

Intervalgradient cut w.r.t. \( \hat{x} \in [\underline{x}, \overline{x}] \): \([d, \bar{d}] := \nabla g ([\underline{x}, \overline{x}])\)

\[
g(\hat{x}) + \min_{d \in [d, \bar{d}]} d^T (x - \hat{x}) \leq 0
\]

Reformulation:

\[
g(\hat{x}) + d^T y^+ - \bar{d}^T y^- \leq 0
\]

\[
x - \hat{x} = y^+ - y^-
\]

\[
0 \leq y_i^+ \leq z_i (\overline{x}_i - \hat{x}_i), \quad i = 1, \ldots, n
\]

\[
0 \leq y_i^- \leq (1 - z_i) (\hat{x}_i - \underline{x}_i), \quad i = 1, \ldots, n
\]

\[
z_i \in \{0, 1\}, \quad i = 1, \ldots, n
\]

- applied to original formulation of MINLP, independent of relaxations
- currently implemented in LaGO with relaxed binary conditions
Further improvements...

- reliable underestimators of nonquadratic nonconvex functions
- support of integer variables
- branching rules
- node selection rules
- ...

Thank you!

http://www.math.hu-berlin.de/~eopt/LaGO
Further improvements...

- reliable underestimators of nonquadratic nonconvex functions
- support of integer variables
- branching rules
- node selection rules
- ...

Thank you!

http://www.math.hu-berlin.de/~eopt/LaGO