Mixed Integer Nonlinear Programming:
Heuristic and Exact Methods

Claudia D’Ambrosio

DEIS, University of Bologna
c.dambrosio@unibo.it

Bordeaux, May 5th, 2011
What is a nonconvex MINLP?

Nonconvex Mixed Integer NonLinear Programming (MINLP).

\[
\begin{align*}
\text{min } & f(x) \\
\text{s.t. } & g(x) \leq 0 \\
\text{ } & x \in X = \{x \mid x \in \mathbb{R}^n, L \leq x \leq U\} \\
\text{ } & x_j \in \mathbb{Z} \quad \forall j \in I
\end{align*}
\]

with \( f(x) : \mathbb{R}^n \rightarrow \mathbb{R} \) and \( g(x) : \mathbb{R}^n \rightarrow \mathbb{R}^m \) are

* continuous
What is a nonconvex MINLP?

**Nonconvex** Mixed Integer NonLinear Programming (MINLP).

\[
\begin{align*}
\text{min } f(x) \\
\quad g(x) & \leq 0 \\
\quad x & \in X = \{ x \mid x \in \mathbb{R}^n, L \leq x \leq U \} \\
\quad x_j & \in \mathbb{Z} \quad \forall j \in I
\end{align*}
\]

with \( f(x) : \mathbb{R}^n \to \mathbb{R} \) and \( g(x) : \mathbb{R}^n \to \mathbb{R}^m \) are

* continuous
* twice differentiable
What is a nonconvex MINLP?

**Nonconvex** Mixed Integer NonLinear Programming (MINLP).

\[
\begin{align*}
\min f(x) \\
g(x) &\leq 0 \\
x &\in X = \{x \mid x \in \mathbb{R}^n, L \leq x \leq U\} \\
x_j &\in \mathbb{Z} \quad \forall j \in I
\end{align*}
\]

with \( f(x) : \mathbb{R}^n \to \mathbb{R} \) and \( g(x) : \mathbb{R}^n \to \mathbb{R}^m \) are

* continuous
* twice differentiable

functions.
What is a nonconvex MINLP?

**Nonconvex** Mixed Integer NonLinear Programming (MINLP).

\[
\begin{align*}
\min & \quad f(x) \\
g(x) & \leq 0 \\
x & \in \mathbf{X} = \{x \mid x \in \mathbb{R}^n, L \leq x \leq U\} \\
x_j & \in \mathbb{Z} \quad \forall j \in I
\end{align*}
\]

with \( f(x) : \mathbb{R}^n \to \mathbb{R} \) and \( g(x) : \mathbb{R}^n \to \mathbb{R}^m \) are
- continuous
- twice differentiable functions.

**NP-hard** problem.
What is a nonconvex MINLP?

Nonconvex Mixed Integer NonLinear Programming (MINLP).

\[
\begin{align*}
\min f(x) \\
g(x) & \leq 0 \\
x & \in X = \{ x \mid x \in \mathbb{R}^n, L \leq x \leq U \} \\
x_j & \in \mathbb{Z} \quad \forall j \in I
\end{align*}
\]

with \( f(x) : \mathbb{R}^n \to \mathbb{R} \) and \( g(x) : \mathbb{R}^n \to \mathbb{R}^m \) are

* continuous

* twice differentiable functions.

- NP-hard problem.
- Local optima \( \neq \) global optima.
C. D’Ambrosio, Linear Approximation Techniques for Mixed Integer Nonlinear Programming: Methods and a Real-world Application, Seminar, Bordeaux, France, January 2011.
MINLP approach: why?

C. D’Ambrosio, Linear Approximation Techniques for Mixed Integer Nonlinear Programming: Methods and a Real-world Application, Seminar, Bordeaux, France, January 2011.

As I said on January, the MILP approach sometimes does not work or it does not provide enough guarantees.
MINLP approach: why?

C. D’Ambrosio, Linear Approximation Techniques for Mixed Integer Nonlinear Programming: Methods and a Real-world Application, Seminar, Bordeaux, France, January 2011.

As I said on January, the MILP approach sometimes does not work or it does not provide enough guarantees.

Part 1: starting from a real-world application, explain heuristic MINLP method.
Part 2: an exact (global optimization) method for MINLP with separable nonconvexities.
Water Distribution Network (WDN) optimal design: choice of a diameter for each pipe, with other fixed design properties (e.g., the topology and pipe lengths).
Water: Introduction

Water Distribution Network (WDN) optimal design: choice of a diameter for each pipe, with other fixed design properties (e.g., the topology and pipe lengths).
Water: Introduction

Water Distribution Network (WDN) optimal design: choice of a diameter for each pipe, with other fixed design properties (e.g., the topology and pipe lengths).
Water: Introduction

Water Distribution Network (WDN) optimal design: choice of a diameter for each pipe, with other fixed design properties (e.g., the topology and pipe lengths).

MINLP problem:

- discrete variables: set of commercially-available diameters;
Water Distribution Network (WDN) optimal design: choice of a diameter for each pipe, with other fixed design properties (e.g., the topology and pipe lengths).

MINLP problem:

- discrete variables: set of commercially-available diameters;
- hydraulic constraints on water flows and pressures;
Water Distribution Network (WDN) optimal design: choice of a diameter for each pipe, with other fixed design properties (e.g., the topology and pipe lengths).

MINLP problem:
- discrete variables: set of commercially-available diameters;
- hydraulic constraints on water flows and pressures;
- minimize the cost (function of the selected diameters).
Main contributions

- First use of mathematical model to solve large-scale real-world instances (metaheuristic approach);
Main contributions

- First use of mathematical model to solve large-scale real-world instances (metaheuristic approach);
- Development of techniques to handle the nonconvexities of the model;
Main contributions

- First use of mathematical model to solve large-scale real-world instances (metaheuristic approach);
- Development of techniques to handle the nonconvexities of the model;
- Implementation of open-source software (COIN-OR);
Main contributions

- First use of mathematical model to solve large-scale real-world instances (metaheuristic approach);
- Development of techniques to handle the nonconvexities of the model;
- Implementation of open-source software (COIN-OR);
- Our approach provides low cost solutions which can be employed as they are in practice (post-process phase needed by the most of the other approaches).
Notation

Sets:
- \( E \): set of pipes;
- \( N \): set of junctions;
- \( S \): set of source junctions \((S \subset N)\);
- \( \delta_+(i) \): set of pipes with tail junction \( i \) \((i \in N)\);
- \( \delta_-(i) \): set of pipes with head junction \( i \) \((i \in N)\).

Parameters for each pipe \( e \in E \):
- \( \text{len}(e) \): length of pipe \( e \);
- \( k(e) \): roughness coefficient of pipe \( e \);
- \( d_{\min}(e), d_{\max}(e) \): min and max diam. of pipe \( e \);
- \( v_{\max}(e) \): max speed of water in pipe \( e \);
- \( \mathcal{D}(e, r), \mathcal{C}(e, r) \): value and cost of the \( r \)th discrete diameter for pipe \( e \) \((r = 1, \ldots, r_e)\).
Parameters for each junction $i \in N \setminus S$:

- $\text{dem}(i) =$ demand at junction $i$;
- $\text{elev}(i) =$ physical elevation of junction $i$;
- $\text{ph}_{\text{min}}(i), \text{ph}_{\text{max}}(i) =$ min and max pressure head at junction $i$.

Parameters for each source junction $i \in S$:

- $h_s(i) =$ fixed hydraulic head of source junction $i$;
Variables

\( Q(e) = \) flow in pipe \( e \ (e \in E) \);
\( H(i) = \) hydraulic head of junction \( i \ (i \in N) \);
\( D(e) = \) diameter of pipe \( e \ (e \in E) \).
A preliminary continuous model

\[ \min \sum_{e \in E} \text{len}(e) \cdot C_e(D(e)) \]
A preliminary continuous model

\[
\min \sum_{e \in E} \text{len}(e) \cdot C_e(D(e)) \\
\sum_{e \in \delta_-(i)} Q(e) - \sum_{e \in \delta_+(i)} Q(e) = \text{dem}(i) \quad (\forall \, i \in N \setminus S)
\]
A preliminary continuous model

\[
\min \sum_{e \in E} len(e) \cdot C_e(D(e))
\]

\[
\sum_{e \in \delta^-(i)} Q(e) - \sum_{e \in \delta^+(i)} Q(e) = dem(i) \quad (\forall \ i \in N \setminus S)
\]

\[-\frac{\pi}{4} v_{\text{max}}(e)D^2(e) \leq Q(e) \leq \frac{\pi}{4} v_{\text{max}}(e)D^2(e) \quad (\forall \ e \in E)\]
A preliminary continuous model

\[
\begin{align*}
\min \ & \sum_{e \in E} len(e) \cdot C_e(D(e)) \\
\sum_{e \in \delta^-(i)} Q(e) - \sum_{e \in \delta^+(i)} Q(e) & = \text{dem}(i) \quad (\forall \ i \in N \setminus S) \\
-\frac{\pi}{4} v_{\max}(e)D^2(e) & \leq Q(e) \leq \frac{\pi}{4} v_{\max}(e)D^2(e) \quad (\forall \ e \in E) \\
H(i) - H(j) & = \frac{\text{sgn}(Q(e))|Q(e)|^{1.852} \cdot 10.7 \cdot \text{len}(e)}{k(e)^{1.852} \cdot D(e)^{4.87}} \quad (\forall \ e = (i,j) \in E)
\end{align*}
\]
A preliminary continuous model

\[
\begin{align*}
\min & \sum_{e \in E} \text{len}(e) \cdot C_e(D(e)) \\
\sum_{e \in \delta^{-}(i)} Q(e) - \sum_{e \in \delta^{+}(i)} Q(e) &= \text{dem}(i) \quad (\forall i \in N \setminus S) \\
- \frac{\pi}{4} v_{\text{max}}(e) D^2(e) &\leq Q(e) \leq \frac{\pi}{4} v_{\text{max}}(e) D^2(e) \quad (\forall e \in E) \\
H(i) - H(j) &= \frac{\text{sgn}(Q(e))|Q(e)|^{1.852} \cdot 10.7 \cdot \text{len}(e)}{k(e)^{1.852} \cdot D(e)^{4.87}} \quad (\forall e = (i, j) \in E) \\
d_{\text{min}}(e) &\leq D(e) \leq d_{\text{max}}(e) \quad (\forall e \in E)
\end{align*}
\]
A preliminary continuous model

\[
\min \sum_{e \in E} len(e) \cdot C_e(D(e))
\]

\[
\sum_{e \in \delta_-(i)} Q(e) - \sum_{e \in \delta_+(i)} Q(e) = dem(i) \quad (\forall i \in N \setminus S)
\]

\[
- \frac{\pi}{4} v_{max}(e) D^2(e) \leq Q(e) \leq \frac{\pi}{4} v_{max}(e) D^2(e) \quad (\forall e \in E)
\]

\[
H(i) - H(j) = \frac{\text{sgn}(Q(e)) |Q(e)|^{1.852} \cdot 10.7 \cdot \text{len}(e)}{k(e)^{1.852} \cdot D(e)^{4.87}} \quad (\forall e = (i, j) \in E)
\]

\[
d_{min}(e) \leq D(e) \leq d_{max}(e) \quad (\forall e \in E)
\]

\[
ph_{min}(i) + elev(i) \leq H(i) \leq ph_{max}(i) + elev(i) \quad (\forall i \in N \setminus S)
\]
A preliminary continuous model

\[
\begin{align*}
\min & \sum_{e \in E} len(e) \cdot C_e(D(e)) \\
\sum_{e \in \delta_-(i)} Q(e) - & \sum_{e \in \delta_+(i)} Q(e) = dem(i) \quad (\forall i \in N \setminus S) \\
- \frac{\pi}{4} v_{\text{max}}(e) D^2(e) \leq & \ Q(e) \leq \frac{\pi}{4} v_{\text{max}}(e) D^2(e) \quad (\forall e \in E) \\
H(i) - & H(j) = \frac{\text{sgn}(Q(e))|Q(e)|^{1.852} \cdot 10.7 \cdot \text{len}(e)}{k(e)^{1.852} \cdot D(e)^{4.87}} \quad (\forall e = (i, j) \in E) \\
\ d_{\text{min}}(e) \leq & \ D(e) \leq d_{\text{max}}(e) \quad (\forall e \in E) \\
ph_{\text{min}}(i) + & \ elev(i) \leq H(i) \leq ph_{\text{max}}(i) + elev(i) \quad (\forall i \in N \setminus S) \\
H(i) = & \ h_s(i) \quad (\forall i \in S).
\end{align*}
\]
A preliminary continuous model: Issues

- Continuous objective function $C_e(D(e))$?
A preliminary continuous model: Issues

- Continuous objective function $C_e(D(e))$?
- Nondifferentiability! See $\text{sgn}(Q(e))|Q(e)|^{1.852}$
Continuous objective function $C_e(D(e))$?

Nondifferentiability! See $\text{sgn}(Q(e))|Q(e)|^{1.852}$

Discretize and reformulate with $A(e)$ instead of $D(e)$!
Fitting a polynomial to the input discrete cost data to make a smooth working continuous cost function $C_e()$.
Continuous objective function

Fitting a polynomial to the input discrete cost data to make a smooth working continuous cost function $C_e()$. We want to minimize the relative error, for example, our least-squares fit for arc $e$ minimizes:

$$\sum_{r=1}^{r_e} \left[ \frac{C(e, r) - \left( \sum_{j=0}^{d} \beta(j, e) \left( \frac{\pi}{4} \mathcal{D}(e, r)^2 \right)^j \right)}{C(e, r)^2} \right]^2$$

$$= \sum_{r=1}^{r_e} \left[ 1 - \left( \frac{\sum_{j=0}^{d} \beta(j, e) \left( \frac{\pi}{4} \mathcal{D}(e, r)^2 \right)^j}{C(e, r)} \right) \right]^2$$
Smoothing the non-differentiability

Approximate $\text{sgn}(Q(e))|Q(e)|^{1.852}$ near 0 ($x \in [-\delta, +\delta]$) with a smooth function $g(x) = ax + bx^3 + cx^5$ having:

- $f(x) = g(x)$;
Smoothing the non-differentiability

Approximate $\text{sgn}(Q(e))|Q(e)|^{1.852}$ near $0 \ (x \in [-\delta, +\delta])$ with a smooth function $g(x) = ax + bx^3 + cx^5$ having:

- $f(x) = g(x)$;
- $f'(x) = g'(x)$;
Approximate $\text{sgn}(Q(e))|Q(e)|^{1.852}$ near 0 ($x \in [-\delta, +\delta]$) with a smooth function $g(x) = ax + bx^3 + cx^5$ having:

- $f(x) = g(x)$;
- $f'(x) = g'(x)$;
- $f''(x) = g''(x)$;
Smoothing the non-differentiability

Approximate \( \text{sgn}(Q(e))|Q(e)|^{1.852} \) near 0 \((x \in [-\delta, +\delta])\) with a smooth function \(g(x) = ax + bx^3 + cx^5\) having:

- \(f(x) = g(x)\);
- \(f'(x) = g'(x)\);
- \(f''(x) = g''(x)\);
- \(f'''(x) = g'''(x)\);

At \(x = \{-\delta, +\delta\}\)
Smoothing the non-differentiability

Approximate $\text{sgn}(Q(e))|Q(e)|^{1.852}$ near 0 ($x \in [-\delta, +\delta]$) with a smooth function $g(x) = ax + bx^3 + cx^5$ having:

- $f(x) = g(x)$;
- $f'(x) = g'(x)$;
- $f''(x) = g''(x)$;
- $f'''(x) = g'''(x)$;

at $x = \{-\delta, +\delta\}$, i.e.,

$$g(x) = \left(\frac{3\delta^{p-5}}{8} + \frac{1}{8}(p - 1)p\delta^{p-5} - \frac{3}{8}p\delta^{p-5}\right)x^5$$

$$+ \left(-\frac{5\delta^{p-3}}{4} - \frac{1}{4}(p - 1)p\delta^{p-3} + \frac{5}{4}p\delta^{p-3}\right)x^3$$

$$+ \left(\frac{15\delta^{p-1}}{8} + \frac{1}{8}(p - 1)p\delta^{p-1} - \frac{7}{8}p\delta^{p-1}\right)x$$

with $p = 1.852$. 
Smoothing the non-differentiability
Reformulating with the area

Involved constraints:

\[-\nu_{\text{max}}(e)A(e) \leq Q(e) \leq \nu_{\text{max}}(e)A(e) \quad (\forall \ e \in E)\]
Reformulating with the area

Involved constraints:

\[ -v_{max}(e)A(e) \leq Q(e) \leq v_{max}(e)A(e) \quad (\forall e \in E) \]

\[ H(i) - H(j) = \frac{\text{sgn}(Q(e))|Q(e)|^{1.852} \cdot 10.7 \cdot \text{len}(e)}{k(e)^{1.852} \cdot A(e)^{2.435} \cdot (\frac{\pi}{4})^{2.435}} \quad (\forall e = (i, j) \in E) \]
Reformulating with the area

Involved constraints:

\[
H(i) - H(j) = \frac{-v_{max}(e)A(e) \leq Q(e) \leq v_{max}(e)A(e)}{\text{sgn}(Q(e))|Q(e)|^{1.852} \cdot 10.7 \cdot \text{len}(e)} \\
\frac{\pi}{4}d_{min}^2(e) \leq A(e) \leq \frac{\pi}{4}d_{max}^2(e) \\
(\forall e \in E)
\]

\[
\frac{\pi}{4}d_{min}^2(e) \leq A(e) \leq \frac{\pi}{4}d_{max}^2(e) \\
(\forall e = (i, j) \in E)
\]
New set of binary variables:

\[ X(e, r) \quad (e \in E, \; r = 1, \ldots, r_e) \]
Discretizing the area

New set of binary variables:

\[ X(e, r) \quad (e \in E, \ r = 1, \ldots, r_e) \]

if \( X(e, r) \) is equal to 1, the \( r \)-th diameter is selected for pipe \( e \).
Discretizing the area

New set of binary variables:

\[ X(e, r) \quad (e \in E, \ r = 1, \ldots, r_e) \]

if \( X(e, r) \) is equal to 1, the \( r \)-th diameter is selected for pipe \( e \).

New constraints:

\[
A(e) = \sum_{r=1}^{r_e} \frac{\pi}{4} D^2(e, r) X(e, r) \quad (e \in E)
\]
Discretizing the area

New set of binary variables:

\[ X(e, r) \quad (e \in E, \ r = 1, \ldots, r_e) \]

if \( X(e, r) \) is equal to 1, the \( r \)-th diameter is selected for pipe \( e \).

New constraints:

\[
A(e) = \sum_{r=1}^{r_e} \frac{\pi}{4} D^2(e, r) X(e, r) \quad (e \in E)
\]

\[
\sum_{r=1}^{r_e} X(e, r) = 1 \quad (e \in E)
\]
Discretizing the area

New set of binary variables:

\[ X(e, r) \quad (e \in E, \; r = 1, \ldots, r_e) \]

if \( X(e, r) \) is equal to 1, the \( r \)-th diameter is selected for pipe \( e \).

New constraints:

\[ A(e) = \sum_{r=1}^{r_e} \frac{\pi}{4} D^2(e, r) X(e, r) \quad (e \in E) \]

\[ \sum_{r=1}^{r_e} X(e, r) = 1 \quad (e \in E) \]

\[ \{X(e, r) | r = 1, \ldots, r_e\} = \text{SOS of type 1} \quad (e \in E). \]
The MINLP model

\[
\min \sum_{e \in E} \text{len}(e) \cdot C_e(A(e))
\]

\[
\min \sum_{e \in E} \text{len}(e) \sum_{r=1}^{r_e} C(e, r) \cdot X(e, r)
\]

\[
\sum_{e \in \delta_-(i)} Q(e) - \sum_{e \in \delta_+(i)} Q(e) = \text{dem}(i) \quad (\forall i \in N \setminus S)
\]

\[
-H(i) - H(j) \cdot k(e)^{1.852} \cdot A(e)^{2.435} \over 10.7 \cdot \text{len}(e) \cdot \left(\frac{\pi}{4}\right)^{2.435} = \begin{cases} 
Q(e)^{1.852} & \text{if } Q(e) \geq \delta \\
g(Q(e)) & \text{if } -\delta < Q(e) < \delta \\
(-Q(e))^{1.852} & \text{if } Q(e) \leq -\delta 
\end{cases} \quad (\forall e = (i, j) \in E)
\]

\[
A(e) = \sum_{r=1}^{r_e} \frac{\pi}{4} D^2(e, r) X(e, r) \quad (\forall e \in E)
\]

\[
\sum_{r=1}^{r_e} X(e, r) = 1 \quad (\forall e \in E)
\]

\[
\frac{\pi}{4} d_{\text{min}}^2(e) \leq A(e) \leq \frac{\pi}{4} d_{\text{max}}^2(e) \quad (\forall e \in E)
\]

\[
ph_{\text{min}}(i) + \text{elev}(i) \leq H(i) \leq ph_{\text{max}}(i) + \text{elev}(i) \quad (\forall i \in N \setminus S)
\]

\[
H(i) = h_s(i) \quad (\forall i \in S).
\]
Bonmin

- Open-source code for solving general MINLP problems
  (https://projects.coin-or.org/Bonmin);
Bonmin

- Open-source code for solving general MINLP problems (https://projects.coin-or.org/Bonmin);
- Several algorithmic choices: a NLP-based Branch-and-Bound algorithm,
Bonmin

- Open-source code for solving general MINLP problems (https://projects.coin-or.org/Bonmin);
- Several algorithmic choices: a NLP-based Branch-and-Bound algorithm, an outer-approximation decomposition algorithm,
Open-source code for solving general MINLP problems (https://projects.coin-or.org/Bonmin);

Several algorithmic choices: a NLP-based Branch-and-Bound algorithm, an outer-approximation decomposition algorithm, a Quesada and Grossmann’s Branch-and-Cut algorithm,
Bonmin

- Open-source code for solving general MINLP problems (https://projects.coin-or.org/Bonmin);
- Several algorithmic choices: a NLP-based Branch-and-Bound algorithm, an outer-approximation decomposition algorithm, a Quesada and Grossmann’s Branch-and-Cut algorithm, and a hybrid outer-approximation based Branch-and-Cut algorithm;
Bonmin

- Open-source code for solving general MINLP problems ([https://projects.coin-or.org/Bonmin](https://projects.coin-or.org/Bonmin));
- Several algorithmic choices: a NLP-based Branch-and-Bound algorithm, an outer-approximation decomposition algorithm, a Quesada and Grossmann’s Branch-and-Cut algorithm, and a hybrid outer-approximation based Branch-and-Cut algorithm;
- Exact for convex MINLPs;
Bonmin

- Open-source code for solving general MINLP problems (https://projects.coin-or.org/Bonmin);
- Several algorithmic choices: a NLP-based Branch-and-Bound algorithm, an outer-approximation decomposition algorithm, a Quesada and Grossmann’s Branch-and-Cut algorithm, and a hybrid outer-approximation based Branch-and-Cut algorithm;
- Exact for convex MINLPs;
- Heuristic for nonconvex MINLPs.
Bonmin and nonconvex MINLPs

All the algorithms implemented in Bonmin (but the Branch-and-Bound) use Outer Approximation cuts (Duran and Grossman, 1986) which are not exact for nonconvex MINLPs.
Bonmin and nonconvex MINLPs

All the algorithms implemented in Bonmin (but the Branch-and-Bound) use Outer Approximation cuts (Duran and Grossman, 1986) which are not exact for nonconvex MINLPs.

\[
f(x) \leq 0 \quad \rightarrow \quad f(x^*) + \nabla f(x^*)^T (x - x^*) \leq 0
\]

where \( \nabla f(x^*) \) is the Jacobian of \( f(x) \) evaluated at the point \( x^* \).
Bonmin and nonconvex MINLPs

All the algorithms implemented in Bonmin (but the Branch-and-Bound) use Outer Approximation cuts (Duran and Grossman, 1986) which are not exact for nonconvex MINLPs.

\[ f(x) \leq 0 \quad \Rightarrow \quad f(x^*) + \nabla f(x^*)^T (x - x^*) \leq 0 \]

where \( \nabla f(x^*) \) is the Jacobian of \( f(x) \) evaluated at the point \( x^* \).
Branch-and-bound algorithm:
Bonmin branch-and-bound

Branch-and-bound algorithm: solve NLP relaxation at each node of the search tree and branch on variables.
Bonmin branch-and-bound

Branch-and-bound algorithm: solve NLP relaxation at each node of the search tree and branch on variables.
NLP solver used: Ipopt (open-source https://projects.coin-or.org/Ipopt).
Bonmin branch-and-bound

Branch-and-bound algorithm: solve NLP relaxation at each node of the search tree and branch on variables.
NLP solver used:
Ipopt (open-source https://projects.coin-or.org/Ipopt).
It founds a local optima: no valid bound for nonconvex MINLPs.
Branch-and-bound algorithm: solve NLP relaxation at each node of the search tree and branch on variables. NLP solver used: Ipopt (open-source \url{https://projects.coin-or.org/Ipopt}). It founds a local optima: no valid bound for nonconvex MINLPs.

LB = 30

0
Bonmin branch-and-bound

**Branch-and-bound algorithm**: solve NLP relaxation at each node of the search tree and branch on variables.

NLP solver used: Ipopt (open-source [https://projects.coin-or.org/Ipopt](https://projects.coin-or.org/Ipopt)). It founds a local optima: no valid bound for nonconvex MINLPs.

\[
\begin{align*}
\text{LB} &= 30 \\
0 &= x_1 = 1 \quad x_1 = 0
\end{align*}
\]
Branch-and-bound algorithm: solve NLP relaxation at each node of the search tree and branch on variables.

NLP solver used: Ipopt (open-source https://projects.coin-or.org/Ipopt). It founds a local optima: no valid bound for nonconvex MINLPs.

\begin{itemize}
\item \textit{LB} = 30
\item \textit{LB} = 35
\item \textit{x}_1 = 0
\item \textit{x}_1 = 1
\end{itemize}
Bonmin branch-and-bound

Branch-and-bound algorithm: solve NLP relaxation at each node of the search tree and branch on variables.

NLP solver used:
Ipopt (open-source https://projects.coin-or.org/Ipopt).
It founds a local optima: no valid bound for nonconvex MINLPs.

```
0
x_1 = 1  x_1 = 0
LB = 30

1
x_2 = 1  x_2 = 0
LB = 35
```
Bonmin branch-and-bound

Branch-and-bound algorithm: solve NLP relaxation at each node of the search tree and branch on variables.

NLP solver used: Ipopt (open-source https://projects.coin-or.org/Ipopt).
It founds a local optima: no valid bound for nonconvex MINLPs.

\[
\begin{align*}
& x_1 = 1 \\
& x_2 = 1 \\
& z = 26
\end{align*}
\]

\[
\begin{align*}
& x_1 = 0 \\
& x_2 = 0 \\
& z = 26
\end{align*}
\]

\[
\begin{align*}
& x_1 = 1 \\
& x_2 = 1 \\
& z = 26
\end{align*}
\]

\[
\begin{align*}
& x_1 = 0 \\
& x_2 = 0 \\
& z = 26
\end{align*}
\]

\[
\begin{align*}
& x_1 = 1 \\
& x_2 = 1 \\
& z = 26
\end{align*}
\]
Bonmin branch-and-bound

Branch-and-bound algorithm: solve NLP relaxation at each node of the search tree and branch on variables.

NLP solver used: Ipopt (open-source https://projects.coin-or.org/Ipopt). It founds a local optima: no valid bound for nonconvex MINLPs.

```
0
  \( x_1 = 1 \) \( x_1 = 0 \)

1
  LB = 35
  \( x_2 = 1 \) \( x_2 = 0 \)

2
  LB = 30
  \( z = 26 \)

3
  \( \emptyset \)
```
Branch-and-bound algorithm: solve NLP relaxation at each node of the search tree and branch on variables.

NLP solver used: Ipopt (open-source https://projects.coin-or.org/Ipopt).

It founds a local optima: no valid bound for nonconvex MINLPs.
Bonmin branch-and-bound

**Branch-and-bound algorithm:** solve NLP relaxation at each node of the search tree and branch on variables.

**NLP solver used:** Ipopt (open-source [https://projects.coin-or.org/Ipopt](https://projects.coin-or.org/Ipopt)). It founds a local optima: no valid bound for nonconvex MINLPs.

![Branch-and-bound tree diagram](image_url)
Branch-and-bound algorithm: solve NLP relaxation at each node of the search tree and branch on variables.

NLP solver used:
Ipopt (open-source https://projects.coin-or.org/Ipopt). It founds a local optima: no valid bound for nonconvex MINLPs.
Bonmin branch-and-bound

Branch-and-bound algorithm: solve NLP relaxation at each node of the search tree and branch on variables.

NLP solver used: Ipopt (open-source https://projects.coin-or.org/Ipopt).

It founds a local optima: no valid bound for nonconvex MINLPs.

\[
\begin{align*}
\text{LB} = 30 & \quad x_1 = 1 \quad x_2 = 1 \\
\text{LB} = 35 & \quad x_1 = 0 \quad x_2 = 0 \\
\text{LB} = 30 & \quad x_1 = 0 \quad x_1 = 1
\end{align*}
\]

\[
\begin{align*}
\text{LB} = 30 & \quad x_1 = 1 \\
\text{LB} = 35 & \quad x_1 = 0 \\
\text{LB} = 30 & \quad x_1 = 1
\end{align*}
\]

\[
\begin{align*}
z = 26 & \quad \nabla \\
z = 27 & \quad \nabla
\end{align*}
\]
Branch-and-bound algorithm: solve NLP relaxation at each node of the search tree and branch on variables.
NLP solver used:
Ipopt (open-source https://projects.coin-or.org/Ipopt). It founds a local optima: no valid bound for nonconvex MINLPs.
Bonmin branch-and-bound

Branch-and-bound algorithm: solve NLP relaxation at each node of the search tree and branch on variables.

NLP solver used:
Ipopt (open-source https://projects.coin-or.org/Ipopt).
It founds a local optima: no valid bound for nonconvex MINLPs.

![Decision Tree Diagram]

- Node 0: LB = 30
  - Node 1: x₁ = 1, x₂ = 1, LB = 35, z = 26
  - Node 2: x₁ = 1, x₂ = 0, z = 27
- Node 4: x₁ = 0, x₂ = 0, LB = 30, z = 27
- Node 3: x₁ = 0, x₂ = 0, LB = 30, z = 27
- Node 5: x₁ = 1, x₂ = 1, LB = 35, z = 27
Bonmin options for nonconvex MINLPs

Different starting points for root/each node.
Bonmin options for nonconvex MINLPs

Different starting points for root/each node.

\[ \text{LB} = \min(30, 0) \]
Bonmin options for nonconvex MINLPs

Different starting points for root/each node.

\[ \text{LB} = \min(30, 28, 0) \]
Bonmin options for nonconvex MINLPs

Different starting points for root/each node.

\[ \text{LB} = \min(30, 28, 32, 0) \]
Bonmin options for nonconvex MINLPs

Different starting points for root/each node.

\[ \text{LB} = \min(30, 28, 32, 30, 0) \]
Bonmin options for nonconvex MINLPs

Different starting points for root/each node.

\[ LB = \min(30, 28, 32, 30, 23) = 23 \]
Different starting points for root/each node.

\[
LB = \min(30, 28, 32, 30, 23) = 23
\]

\[
x_1 = 1 \quad \text{or} \quad x_1 = 0
\]
Bonmin options for nonconvex MINLPs

Different starting points for root/each node.

$$LB = \min(30, 28, 32, 30, 23) = 23$$

Diagram:

- Node 0 with $x_1 = 1$
- Node 1 with $x_1 = 0$

$$LB = \min(35,$$
Bonmin options for nonconvex MINLPs

Different starting points for root/each node.

\[ LB = \min(30, 28, 32, 30, 23) = 23 \]

\[ x_1 = 1 \quad \text{and} \quad x_1 = 0 \]

\[ LB = \min(35, 24, \ldots) \]
Different starting points for root/each node.

\[ LB = \min(30, 28, 32, 30, 23) = 23 \]

\[ LB = \min(35, 24, 28, \ldots) \]
Bonmin options for nonconvex MINLPs

Different starting points for root/each node.

\[ \text{LB} = \min(30, 28, 32, 30, 23) = 23 \]

\[ x_1 = 1 \quad \text{and} \quad x_1 = 0 \]

\[ \text{LB} = \min(35, 24, 28, 24, \ldots) \]
Different starting points for root/each node.

\[ \text{LB} = \min(30, 28, 32, 30, 23) = 23 \]

\[ x_1 = 1 \quad \text{or} \quad x_1 = 0 \]

\[ \text{LB} = \min(35, 24, 28, 24, 30) = 24 \]
Different starting points for root/each node.

\[ \text{LB} = \min(30, 28, 32, 30, 23) = 23 \]

\[ x_1 = 1 \quad x_1 = 0 \]

\[ \text{LB} = \min(35, 24, 28, 24, 30) = 24 \]

\[ x_2 = 1 \quad x_2 = 0 \]
Bonmin options for nonconvex MINLPs

Different starting points for root/each node.

LB = min(30, 28, 32, 30, 23) = 23

LB = min(35, 24, 28, 24, 30) = 24

z = 26!!!!!!
Bonmin options for nonconvex MINLPs

Different starting points for root/each node.

\[
\begin{align*}
\text{LB} &= \min(30, 28, 32, 30, 23) = 23 \\
0 & \quad x_1 = 1 \quad x_1 = 0 \\
1 & \quad x_2 = 1 \quad x_2 = 0 \\
2 & \quad z = 26 !!!!!! \\
3 & \quad \emptyset
\end{align*}
\]
Bonmin options for nonconvex MINLPs

Different starting points for root/each node.

LB = \min(30, 28, 32, 30, 23) = 23

LB = \min(35, 24, 28, 24, 30) = 24

z = 26!!!!!!

z = 27
Bonmin options for nonconvex MINLPs

Different starting points for root/each node.

\[ \text{LB} = \min(30, 28, 32, 30, 23) = 23 \]

\[ \text{LB} = \min(35, 24, 28, 24, 30) = 24 \]

\[ x_1 = 1 \quad x_1 = 0 \]

\[ x_2 = 1 \quad x_2 = 0 \]

\[ z = 26 \quad z = 27 \]

Still not a valid LB!
Bonmin options for nonconvex MINLPs

Fathom only if \( z \leq LB + \Delta z \) with \( \Delta z < 0 \), i.e., continue branching even if the solution value to the current node is worse than the best-known solution.
Bonmin options for nonconvex MINLPs

Fathom only if $z \leq LB + \Delta z$ with $\Delta z < 0$, i.e., continue branching even if the solution value to the current node is worse than the best-known solution.

\[
LB = 30
\]

\[
0
\]
Bonmin options for nonconvex MINLPs

Fathom only if $z \leq LB + \Delta z$ with $\Delta z < 0$, i.e., continue branching even if the solution value to the current node is worse than the best-known solution.

\[ LB = 30 \]

\[ 0 \]

\[ x_1 = 0 \quad x_1 = 1 \]
Fathom only if $z \leq LB + \Delta z$ with $\Delta z < 0$, i.e., continue branching even if the solution value to the current node is worse than the best-known solution.

![Diagram]

\begin{align*}
LB &= 30 \\
0 &
\begin{cases}
  x_1 = 0 \\
  x_1 = 1
\end{cases}
\begin{cases}
  z = 27
\end{cases}
\end{align*}
Bonmin options for nonconvex MINLPs

Fathom only if \( z \leq LB + \Delta z \) with \( \Delta z < 0 \), i.e., continue branching even if the solution value to the current node is worse than the best-known solution.

Let \( \Delta z \) be equal -10. Do not fathom node 2 if \( z > LB + \Delta z \), i.e. \( 27 > 35 - 10 = 25! \)
Fathom only if $z \leq LB + \Delta z$ with $\Delta z < 0$, i.e., continue branching even if the solution value to the current node is worse than the best-known solution.

Let $\Delta z$ be equal -10. Do not fathom node 2 if $z > LB + \Delta z$, i.e. $27 > 35 - 10 = 25$!

If $|\Delta z|$ is too big, few nodes are fathomed, extreme case: complete enumeration!
1.1 Ad-hoc definition of the cutoff_decr option value ($\Delta z$).
**Bonmin modifications**

1.1 Ad-hoc definition of the cutoff\_decr option value ($\Delta z$).

1.2 Properly evaluating the objective value of integer feasible solutions through the definition of 2 objective functions: LB and UB objective function;
Bonmin modifications: ad-hoc $\Delta z$

Ad-hoc definition of the $\Delta z$ option value: If the values of the local optima of a specific instance differs of a large amount, then $|\Delta z|$ should be large.
Ad-hoc definition of the $\Delta z$ option value:
If the values of the local optima of a specific instance differs of a large amount, then $|\Delta z|$ should be large.
If the values of the local optima of a specific instance are similar, then $|\Delta z|$ should be small.
Bonmin modifications: ad-hoc $\Delta z$

Ad-hoc definition of the $\Delta z$ option value:
If the values of the local optima of a specific instance differs of a large amount, then $|\Delta z|$ should be large.
If the values of the local optima of a specific instance are similar, then $|\Delta z|$ should be small.
Solution: collect statistics on the values of the local optima solving with multiple starting points the root, then set:

$$\Delta z := -V \cdot \begin{cases} .02, & \text{if } \sigma < .1; \\ .05, & \text{if } \sigma \geq .1. \end{cases}$$

where $V$ is the average of the root node continuous values, $\sigma \in [0, 1]$ is the coefficient of variation (standard deviation divided by the mean) of those values.
Bonmin modifications: 2 objective functions

Lower bound at each branch-and-bound node found using the continuous objective function

$$\min \sum_{e \in E} len(e) \cdot C_e(A(e))$$

The continuous objective function value can differ from the actual value of an integer feasible solution

$$\min \sum_{e \in E} len(e) \sum_{r=1}^{r_e} c(e, r) \cdot X(e, r).$$

When an integer feasible solution is found, it is evaluated with respect to the discrete objective function. In this way the incumbent is updated.
## Computational results

### Instances.

<table>
<thead>
<tr>
<th>name</th>
<th>junctions</th>
<th>reservoirs</th>
<th>pipes</th>
<th>duplicates</th>
<th>diameters</th>
<th>unit cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>shamir</td>
<td>7</td>
<td>1</td>
<td>8</td>
<td>–</td>
<td>14</td>
<td>$</td>
</tr>
<tr>
<td>hanoi</td>
<td>32</td>
<td>1</td>
<td>34</td>
<td>–</td>
<td>6</td>
<td>$</td>
</tr>
<tr>
<td>blacksburg</td>
<td>31</td>
<td>1</td>
<td>35</td>
<td>–</td>
<td>11</td>
<td>$</td>
</tr>
<tr>
<td>New York</td>
<td>20</td>
<td>1</td>
<td>21</td>
<td>21</td>
<td>12</td>
<td>$</td>
</tr>
<tr>
<td>foss_poly_0</td>
<td>37</td>
<td>1</td>
<td>58</td>
<td>–</td>
<td>7</td>
<td>lira</td>
</tr>
<tr>
<td>foss_iron</td>
<td>37</td>
<td>1</td>
<td>58</td>
<td>–</td>
<td>13</td>
<td>€</td>
</tr>
<tr>
<td>foss_poly_1</td>
<td>37</td>
<td>1</td>
<td>58</td>
<td>–</td>
<td>22</td>
<td>€</td>
</tr>
<tr>
<td>pescara</td>
<td>71</td>
<td>3</td>
<td>99</td>
<td>–</td>
<td>13</td>
<td>€</td>
</tr>
<tr>
<td>modena</td>
<td>272</td>
<td>4</td>
<td>317</td>
<td>–</td>
<td>13</td>
<td>€</td>
</tr>
</tbody>
</table>
### Computational results

Characteristics of the 50 continuous solutions at the root node.

<table>
<thead>
<tr>
<th>Location</th>
<th>mean</th>
<th>% dev. first</th>
<th>% dev. min</th>
<th>% dev. max</th>
<th>std dev</th>
<th>coeff var</th>
</tr>
</thead>
<tbody>
<tr>
<td>shamir</td>
<td>401,889.00</td>
<td>-4.880</td>
<td>-4.880</td>
<td>59.707</td>
<td>37,854.70</td>
<td>0.0941920</td>
</tr>
<tr>
<td>hanoi</td>
<td>6,134,520.00</td>
<td>-0.335</td>
<td>-1.989</td>
<td>2.516</td>
<td>91,833.70</td>
<td>0.0149700</td>
</tr>
<tr>
<td>blacksburg</td>
<td>114,163.00</td>
<td>1.205</td>
<td>-0.653</td>
<td>2.377</td>
<td>861.92</td>
<td>0.0075499</td>
</tr>
<tr>
<td>New York</td>
<td>82,646,700.00</td>
<td>0.605</td>
<td>-47.928</td>
<td>31.301</td>
<td>16,682,600.00</td>
<td>0.2018540</td>
</tr>
<tr>
<td>foss_poly_0</td>
<td>68,601,200.00</td>
<td>-1.607</td>
<td>-1.748</td>
<td>15.794</td>
<td>2,973,570.00</td>
<td>0.0433457</td>
</tr>
<tr>
<td>foss_iron</td>
<td>182,695.00</td>
<td>-2.686</td>
<td>-2.686</td>
<td>61.359</td>
<td>16,933.80</td>
<td>0.0926891</td>
</tr>
<tr>
<td>foss_poly_1</td>
<td>32,195.40</td>
<td>26.186</td>
<td>-17.193</td>
<td>42.108</td>
<td>4,592.63</td>
<td>0.1426490</td>
</tr>
<tr>
<td>pescara</td>
<td>1,937,180.00</td>
<td>-6.311</td>
<td>-6.596</td>
<td>54.368</td>
<td>274,956.00</td>
<td>0.1419370</td>
</tr>
<tr>
<td>modena</td>
<td>2,559,350.00</td>
<td>-0.254</td>
<td>-0.396</td>
<td>9.191</td>
<td>38,505.80</td>
<td>0.0150452</td>
</tr>
</tbody>
</table>
Computational results for the MINLP model (part 1). Time limit 7200 seconds.

<table>
<thead>
<tr>
<th>Location</th>
<th>$v_{fit}(x^{best})$</th>
<th>$v_{disc}(\bar{x}^{best})$</th>
<th>% dev. $v_{fit}(\bar{x}^{best})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>shamir</td>
<td>400,749.77</td>
<td>419,000.00</td>
<td>0.000</td>
</tr>
<tr>
<td>hanoi</td>
<td>6,109,840.00</td>
<td>6,109,620.90</td>
<td>0.000</td>
</tr>
<tr>
<td>blacksburg</td>
<td>118,251.06</td>
<td>118,251.09</td>
<td>0.000</td>
</tr>
<tr>
<td>New York</td>
<td>39,569,920.48</td>
<td>39,307,799.72</td>
<td>0.550</td>
</tr>
<tr>
<td>foss_poly_0</td>
<td>69,161,700.00</td>
<td>70,680,507.90</td>
<td>0.000</td>
</tr>
<tr>
<td>foss_iron</td>
<td>179,134.00</td>
<td>178,494.14</td>
<td>0.015</td>
</tr>
<tr>
<td>foss_poly_1</td>
<td>29,016.60</td>
<td>29,117.04</td>
<td>0.000</td>
</tr>
<tr>
<td>pescara</td>
<td>1,850,720.00</td>
<td>1,820,263.72</td>
<td>0.193</td>
</tr>
<tr>
<td>modena</td>
<td>2,609,550.00</td>
<td>2,576,589.00</td>
<td>0.341</td>
</tr>
</tbody>
</table>
Computational results

Computational results for the MINLP model (part 2). Time limit 7200 seconds.

<table>
<thead>
<tr>
<th></th>
<th>$v_{\text{disc}}(\bar{x}^{\text{best}})$</th>
<th>time</th>
<th>v_{\text{disc}}(\bar{x}^{\text{first}})</th>
<th># fit</th>
<th># true</th>
</tr>
</thead>
<tbody>
<tr>
<td>shamir</td>
<td>419,000.00</td>
<td>1</td>
<td>0.000</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>hanoi</td>
<td>6,109,620.90</td>
<td>191</td>
<td>0.000</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>blacksburg</td>
<td>118,251.09</td>
<td>2,018</td>
<td>0.178</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>New York</td>
<td>39,307,799.72</td>
<td>5</td>
<td>0.000</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>foss_poly_0</td>
<td>70,680,507.90</td>
<td>41</td>
<td>0.000</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>foss_iron</td>
<td>178,494.14</td>
<td>464</td>
<td>0.000</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>foss_poly_1</td>
<td>29,117.04</td>
<td>2,589</td>
<td>0.119</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>pescara</td>
<td>1,820,263.72</td>
<td>2,084</td>
<td>0.724</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>modena</td>
<td>2,576,589.00</td>
<td>3,935</td>
<td>0.055</td>
<td>4</td>
<td>3</td>
</tr>
</tbody>
</table>
Baron: considered the best solver for global optimization.
Baron: considered the best solver for global optimization. It uses:

- under and over estimators to compute valid LB for nonconvex MINLPs;
**Baron**: considered the best solver for global optimization. It uses:

- under and over estimators to compute valid LB for nonconvex MINLPs;
- spatial Branch-and-Bound to improve estimation quality and enforce integrality.
Baron: considered the best solver for global optimization. It uses:

- under and over estimators to compute valid LB for nonconvex MINLPs;
- spatial Branch-and-Bound to improve estimation quality and enforce integrality.

It provides (possibly) the global solution of nonconvex MINLPs (a lower bound otherwise).
Baron: considered the best solver for global optimization. It uses:

- under and over estimators to compute valid LB for nonconvex MINLPs;
- spatial Branch-and-Bound to improve estimation quality and enforce integrality.

It provides (possibly) the global solution of nonconvex MINLPs (a lower bound otherwise). Used to measure the quality of the proposed solutions.
Computing valid lower bounds with Baron

Computational results for the MINLP model comparing Baron and Bonmin

<table>
<thead>
<tr>
<th></th>
<th>UB (2h)</th>
<th>LB (2h)</th>
<th>UB (12h)</th>
<th>LB (12h)</th>
<th>Bonmin % gap</th>
</tr>
</thead>
<tbody>
<tr>
<td>shamir</td>
<td>419,000.00</td>
<td>419,000.00</td>
<td>419,000.00</td>
<td>419,000.00</td>
<td>0.00</td>
</tr>
<tr>
<td>hanoi</td>
<td>6,309,727.80</td>
<td>5,643,490.00</td>
<td>6,219,567.80</td>
<td>5,783,950.00</td>
<td>5.63</td>
</tr>
<tr>
<td>blacksburg</td>
<td>n.a.</td>
<td>55,791.90</td>
<td>n.a.</td>
<td>105,464.00</td>
<td>12.12</td>
</tr>
<tr>
<td>New York</td>
<td>43,821,000.00</td>
<td>29,174,000.00</td>
<td>43,821,000.00</td>
<td>29,174,000.00</td>
<td>34.74</td>
</tr>
<tr>
<td>foss_poly_0</td>
<td>n.a.</td>
<td>64,787,300.00</td>
<td>n.a.</td>
<td>64,787,300.00</td>
<td>9.10</td>
</tr>
<tr>
<td>foss_iron</td>
<td>n.a.</td>
<td>170,580.00</td>
<td>n.a.</td>
<td>170,580.00</td>
<td>4.64</td>
</tr>
<tr>
<td>foss_poly_1</td>
<td>n.a.</td>
<td>25,308.20</td>
<td>n.a.</td>
<td>25,308.20</td>
<td>15.05</td>
</tr>
<tr>
<td>pescara</td>
<td>n.a.</td>
<td>1,512,640.00</td>
<td>n.a.</td>
<td>1,512,640.00</td>
<td>20.34</td>
</tr>
<tr>
<td>modena</td>
<td>n.a.</td>
<td>2,073,050.00</td>
<td>n.a.</td>
<td>2,073,050.00</td>
<td>24.29</td>
</tr>
</tbody>
</table>
MILP solutions

The nonlinear parts of the models: fitted objective function and Hazen-Williams equations.
The nonlinear parts of the models: fitted objective function and Hazen-Williams equations.

- Consider only the discrete objective function;
MILP solutions

The nonlinear parts of the models: fitted objective function and Hazen-Williams equations.

- Consider only the discrete objective function;
- Linearize the Hazen-Williams equations (for each Diameter → piecewise linear approximation of univariate functions).
The nonlinear parts of the models: fitted objective function and Hazen-Williams equations.

- Consider only the discrete objective function;
- Linearize the Hazen-Williams equations (for each Diameter → piecewise linear approximation of univariate functions).

Additional continuous and binary variables and additional constraints (proportional to the quality of the approximation).
The nonlinear parts of the models: fitted objective function and Hazen-Williams equations.

- Consider only the discrete objective function;
- Linearize the Hazen-Williams equations (for each Diameter $\rightarrow$ piecewise linear approximation of univariate functions).

Additional continuous and binary variables and additional constraints (proportional to the quality of the approximation).

When all the diameters/areas have been selected, the objective function is a constant and we need complete enumeration to find the values of the other variables!
The nonlinear parts of the models: fitted objective function and Hazen-Williams equations.

- Consider only the discrete objective function;
- Linearize the Hazen-Williams equations (for each Diameter → piecewise linear approximation of univariate functions).

Additional continuous and binary variables and additional constraints (proportional to the quality of the approximation). When all the diameters/areas have been selected, the objective function is a constant and we need complete enumeration to find the values of the other variables!

_shamir_: the same solution found with the MINLP approach.
MILP solutions

The nonlinear parts of the models: fitted objective function and Hazen-Williams equations.

- Consider only the discrete objective function;
- Linearize the Hazen-Williams equations (for each Diameter $\rightarrow$ piecewise linear approximation of univariate functions).

Additional continuous and binary variables and additional constraints (proportional to the quality of the approximation).
When all the diameters/areas have been selected, the objective function is a constant and we need complete enumeration to find the values of the other variables!

shamir: the same solution found with the MINLP approach.
hanoi: a worse and slightly infeasible solution in 40 CPU minutes.
MILP solutions

The nonlinear parts of the models: fitted objective function and Hazen-Williams equations.

- Consider only the discrete objective function;
- Linearize the Hazen-Williams equations (for each Diameter → piecewise linear approximation of univariate functions).

Additional continuous and binary variables and additional constraints (proportional to the quality of the approximation).

When all the diameters/areas have been selected, the objective function is a constant and we need complete enumeration to find the values of the other variables!

- shamir: the same solution found with the MINLP approach.
- hanoi: a worse and slightly infeasible solution in 40 CPU minutes.
- blacksburg and New York: a worse solution in 48 CPU hours!!!
The nonlinear parts of the models: fitted objective function and Hazen-Williams equations.

- Consider only the discrete objective function;
- Linearize the Hazen-Williams equations (for each Diameter → piecewise linear approximation of univariate functions).

Additional continuous and binary variables and additional constraints (proportional to the quality of the approximation).

When all the diameters/areas have been selected, the objective function is a constant and we need complete enumeration to find the values of the other variables!

**shamir**: the same solution found with the MINLP approach.
**hanoi**: a worse and slightly infeasible solution in 40 CPU minutes.
**blacksburg** and **New York**: a worse solution in 48 CPU hours!!!

Real-world instances: even worse results.
Practical use of MINLP solutions
Global Optimization.

Aim: finding a global optimum of the non-convex MINLP problem.
Global Optimization.

Aim: finding a global optimum of the non-convex MINLP problem.

Real-world application: a confidential problem within IBM.
The class of problems

\[
\min \sum_{j \in N} C_j x_j \\
\text{subject to} \quad f(x) \leq 0 ; \\
r_i(x) + \sum_{k \in H(i)} g_{ik}(x_k) \leq 0 , \ \forall i \in M ; \\
L_j \leq x_j \leq U_j , \ \forall j \in N ; \\
x_j \text{ integer}, \ \forall j \in I ,
\]

where:
\( f : \mathbb{R}^n \rightarrow \mathbb{R}^p \) and \( r_i : \mathbb{R}^n \rightarrow \mathbb{R} \ \forall i \in M \), are convex functions,
\( H(i) \subseteq N \ \forall i \in M \),
the \( g_{ik} : \mathbb{R} \rightarrow \mathbb{R} \) are non-convex univariate function \( \forall i \in M \),
and \( I \subseteq N \).
For \( j \in H \) we assume that the bounds are finite.
The SC-MINLP (Sequential Convex MINLP) framework
The Lower Bounding problem \( Q \): step 1

For simplicity, let us consider a term of the form \( g(x_k) := g_{ik}(x_k) \): 
\( g : \mathbb{R} \rightarrow \mathbb{R} \) is a univariate non-convex function of \( x_k \), for some \( k \) \((1 \leq k \leq n)\).
The Lower Bounding problem \( Q \): step 1

For simplicity, let us consider a term of the form \( g(x_k) := g_{ik}(x_k) \):
\[ g : \mathbb{R} \to \mathbb{R} \]
is a univariate non-convex function of \( x_k \), for some \( k \)
\( (1 \leq k \leq n) \).

Automatically detect the concavity/convexity intervals or piecewise definition:
\[
[P_{p-1}, P_p] := \text{the } p\text{-th subinterval of the domain of } g \ (p \in \{1 \ldots \bar{p}\}) ;
\]
The Lower Bounding problem $Q$: step 1

For simplicity, let us consider a term of the form $g(x_k) := g_{ik}(x_k)$:

$g : \mathbb{R} \rightarrow \mathbb{R}$ is a univariate non-convex function of $x_k$, for some $k (1 \leq k \leq n)$. 

![Graph of g(x) showing concavity/convexity intervals]

Automatically detect the concavity/convexity intervals or piecewise definition:

$[P_{p-1}, P_p] :=$ the $p$-th subinterval of the domain of $g$ ($p \in \{1 \ldots \bar{p}\}$);

$\tilde{H} :=$ the set of indices of subintervals on which $g$ is convex;
The Lower Bounding problem \( Q \): step 1

For simplicity, let us consider a term of the form \( g(x_k) := g_{ik}(x_k) \): 
\( g : \mathbb{R} \rightarrow \mathbb{R} \) is a univariate non-convex function of \( x_k \), for some \( k \) (\( 1 \leq k \leq n \)).

Automatically detect the concavity/convexity intervals or piecewise definition:
\[ [P_{p-1}, P_p] := \text{the } p\text{-th subinterval of the domain of } g \ (p \in \{1 \ldots \overline{p}\}); \]
\( \tilde{H} := \text{the set of indices of subintervals on which } g \text{ is convex}; \)
\( \hat{H} := \text{the set of indices of subintervals on which } g \text{ is concave}. \)
Replace the term $g(x_k)$ with:

$$\sum_{p=1}^{\bar{p}} g(P_{p-1} + \delta_p) - \sum_{p=1}^{\bar{p}-1} g(P_p).$$
Replace the term $g(x_k)$ with:

$$\sum_{p=1}^{\bar{p}} g(P_{p-1} + \delta_p) - \sum_{p=1}^{\bar{p}-1} g(P_p),$$

and we include the following set of new constraints:

$$P_0 + \sum_{p=1}^{\bar{p}} \delta_p - x_k = 0;$$
The Lower Bounding problem $Q$: step 2

Replace the term $g(x_k)$ with:

$$
\sum_{p=1}^{\bar{p}} g(P_{p-1} + \delta_p) - \sum_{p=1}^{\bar{p}-1} g(P_p),
$$

and we include the following set of new constraints:

$$
P_0 + \sum_{p=1}^{\bar{p}} \delta_p - x_k = 0; \\
\delta_p - (P_p - P_{p-1})z_p \geq 0, \forall p \in \tilde{H} \cup \hat{H}; \\
\delta_p - (P_p - P_{p-1})z_{p-1} \leq 0, \forall p \in \tilde{H} \cup \hat{H}; \\
0 \leq \delta_p \leq P_p - P_{p-1}, \forall p \in \{1, \ldots, \bar{p}\};
$$

with two dummy variables $z_0 := 1$ and $z_{\bar{p}} := 0$ and two new sets of variables $z_p$ (binary) and $\delta_p$ (continuous).
The Lower Bounding problem \( Q \): step 2

Replace the term \( g(x_k) \) with:

\[
\sum_{p=1}^{\bar{p}} g(P_{p-1} + \delta_p) - \sum_{p=1}^{\bar{p}-1} g(P_p),
\]

and we include the following set of new constraints:

\[
P_0 + \sum_{p=1}^{\bar{p}} \delta_p - x_k = 0 ;
\]
\[
\delta_p - (P_p - P_{p-1})z_p \geq 0 , \forall p \in \tilde{H} \cup \hat{H} ;
\]
\[
\delta_p - (P_p - P_{p-1})z_{p-1} \leq 0 , \forall p \in \tilde{H} \cup \hat{H} ;
\]
\[
0 \leq \delta_p \leq P_p - P_{p-1} , \forall p \in \{1, \ldots, \bar{p}\};
\]

with two dummy variables \( z_0 := 1 \) and \( z_{\bar{p}} := 0 \) and two new sets of variables \( z_p \) (binary) and \( \delta_p \) (continuous).

If \( x_k \in [P_{p^*-1}, P_{p^*}] \)
The Lower Bounding problem $Q$: step 2

Replace the term $g(x_k)$ with:

$$\sum_{p=1}^{\bar{p}} g(P_{p-1} + \delta_p) - \sum_{p=1}^{\bar{p}-1} g(P_p),$$

and we include the following set of new constraints:

$$P_0 + \sum_{p=1}^{\bar{p}} \delta_p - x_k = 0;$$
$$\delta_p - (P_p - P_{p-1})z_p \geq 0, \forall p \in \tilde{H} \cup \hat{H};$$
$$\delta_p - (P_p - P_{p-1})z_{p-1} \leq 0, \forall p \in \tilde{H} \cup \hat{H};$$
$$0 \leq \delta_p \leq P_p - P_{p-1}, \forall p \in \{1, \ldots, \bar{p}\};$$

with two dummy variables $z_0 := 1$ and $z_{\bar{p}} := 0$ and two new sets of variables $z_p$ (binary) and $\delta_p$ (continuous).

If $x_k \in [P_{p^*-1}, P_{p^*}]$ then $z_p = 1$ for $p = 1, \ldots, p^*-1$ and
The Lower Bounding problem \( Q \): step 2

Replace the term \( g(x_k) \) with:

\[
\sum_{p=1}^{\bar{p}} g(P_{p-1} + \delta_p) - \sum_{p=1}^{\bar{p}-1} g(P_p),
\]

and we include the following set of new constraints:

\[
P_0 + \sum_{p=1}^{\bar{p}} \delta_p - x_k = 0;
\]

\[
\delta_p - (P_p - P_{p-1})z_p \geq 0, \quad \forall p \in \tilde{\mathcal{H}} \cup \hat{\mathcal{H}};
\]

\[
\delta_p - (P_p - P_{p-1})z_{p-1} \leq 0, \quad \forall p \in \tilde{\mathcal{H}} \cup \hat{\mathcal{H}};
\]

\[
0 \leq \delta_p \leq P_p - P_{p-1}, \quad \forall p \in \{1, \ldots, \bar{p}\};
\]

with two dummy variables \( z_0 := 1 \) and \( z_{\bar{p}} := 0 \) and two new sets of variables \( z_p \) (binary) and \( \delta_p \) (continuous).

If \( x_k \in [P_{p^*-1}, P_{p^*}] \) then \( z_p = 1 \) for \( p = 1, \ldots, p^* - 1 \) and \( z_p = 0 \) for \( p = p^*, \ldots, \bar{p} \).
Replace the term \( g(x_k) \) with:

\[
\sum_{p=1}^{\bar{p}} g(P_{p-1} + \delta_p) - \sum_{p=1}^{\bar{p}-1} g(P_p) ,
\]

and we include the following set of new constraints:

\[
P_0 + \sum_{p=1}^{\bar{p}} \delta_p - x_k = 0 ;
\]

\[
\delta_p - (P_p - P_{p-1})z_p \geq 0 , \ \forall p \in \tilde{H} \cup \hat{H} ;
\]

\[
\delta_p - (P_p - P_{p-1})z_{p-1} \leq 0 , \ \forall p \in \tilde{H} \cup \hat{H} ;
\]

\[
0 \leq \delta_p \leq P_p - P_{p-1} , \ \forall p \in \{1, \ldots, \bar{p}\} ;
\]

with two dummy variables \( z_0 := 1 \) and \( z_{\bar{p}} := 0 \) and two new sets of variables \( z_p \) (binary) and \( \delta_p \) (continuous).

If \( x_k \in [P_{p^*-1}, P_{p^*}] \) then \( z_p = 1 \) for \( p = 1, \ldots, p^* - 1 \) and \( z_p = 0 \) for \( p = p^*, \ldots, \bar{p} \), \( \delta_p = P_p - P_{p-1} \) for \( p = 1, \ldots, p^* - 1 \),
The Lower Bounding problem $Q$: step 2

Replace the term $g(x_k)$ with:

$$\sum_{p=1}^{\bar{p}} g(P_{p-1} + \delta_p) - \sum_{p=1}^{\bar{p}-1} g(P_p),$$

and we include the following set of new constraints:

$$P_0 + \sum_{p=1}^{\bar{p}} \delta_p - x_k = 0;$$
$$\delta_p - (P_p - P_{p-1}) z_p \geq 0, \forall p \in \tilde{H} \cup \hat{H};$$
$$\delta_p - (P_p - P_{p-1}) z_{p-1} \leq 0, \forall p \in \tilde{H} \cup \hat{H};$$
$$0 \leq \delta_p \leq P_p - P_{p-1}, \forall p \in \{1, \ldots, \bar{p}\};$$

with two dummy variables $z_0 := 1$ and $z_{\bar{p}} := 0$ and two new sets of variables $z_p$ (binary) and $\delta_p$ (continuous).

If $x_k \in [P_{p^*-1}, P_{p^*}]$ then $z_p = 1$ for $p = 1, \ldots, p^* - 1$ and $z_p = 0$ for $p = p^*, \ldots, \bar{p}$, $\delta_p = P_p - P_{p-1}$ for $p = 1, \ldots, p^* - 1$, $0 \leq \delta_{p^*} \leq P_{p^*} - P_{p^*-1}$, and
The Lower Bounding problem $Q$: step 2

Replace the term $g(x_k)$ with:

$$\sum_{p=1}^{\bar{p}} g(P_{p-1} + \delta_p) - \sum_{p=1}^{\bar{p}-1} g(P_p),$$

and we include the following set of new constraints:

$$P_0 + \sum_{p=1}^{\bar{p}} \delta_p - x_k = 0;$$

$$\delta_p - (P_p - P_{p-1})z_p \geq 0, \quad \forall p \in \tilde{H} \cup \hat{H};$$

$$\delta_p - (P_p - P_{p-1})z_{p-1} \leq 0, \quad \forall p \in \tilde{H} \cup \hat{H};$$

$$0 \leq \delta_p \leq P_p - P_{p-1}, \quad \forall p \in \{1, \ldots, \bar{p}\};$$

with two dummy variables $z_0 := 1$ and $z_{\bar{p}} := 0$ and two new sets of variables $z_p$ (binary) and $\delta_p$ (continuous).

If $x_k \in [P_{p^*-1}, P_{p^*}]$ then $z_p = 1$ for $p = 1, \ldots, p^* - 1$ and $z_p = 0$ for $p = p^*, \ldots, \bar{p}$, $\delta_p = P_p - P_{p-1}$ for $p = 1, \ldots, p^* - 1$, $0 \leq \delta_{p^*} \leq P_{p^*} - P_{p^*-1}$, and $\delta_p = 0$ for $p = p^* + 1, \ldots, \bar{p}$. 
0 ≤ δ₁ ≤ (1 − 0) = 1, 0 ≤ δ₂ ≤ (2 − 1) = 1, 0 ≤ δ₃ ≤ (4 − 2) = 2.
The Lower Bounding problem $Q$: step 2

\[ 0 \leq \delta_1 \leq (1 - 0) = 1, \ 0 \leq \delta_2 \leq (2 - 1) = 1, \ 0 \leq \delta_3 \leq (4 - 2) = 2. \]

If $x = 1.5$ then $\delta_1 = 1, \ \delta_2 = 0.5, \ \delta_3 = 0.$
The Lower Bounding problem $Q$: step 2

$0 \leq \delta_1 \leq (1 - 0) = 1$, $0 \leq \delta_2 \leq (2 - 1) = 1$, $0 \leq \delta_3 \leq (4 - 2) = 2$.
If $x = 1.5$ then $\delta_1 = 1$, $\delta_2 = 0.5$, $\delta_3 = 0$. Then

$g(x) =$
The Lower Bounding problem $Q$: step 2

\[ 0 \leq \delta_1 \leq (1 - 0) = 1, \quad 0 \leq \delta_2 \leq (2 - 1) = 1, \quad 0 \leq \delta_3 \leq (4 - 2) = 2. \]

If \( x = 1.5 \) then \( \delta_1 = 1, \quad \delta_2 = 0.5, \quad \delta_3 = 0. \) Then

\[ g(x) = \sum_{p=1}^{\overline{p}} g(P_{p-1} + \delta_p) - \sum_{p=1}^{\overline{p}-1} g(P_p) \]
The Lower Bounding problem $Q$: step 2

\[ 0 \leq \delta_1 \leq (1 - 0) = 1, \ 0 \leq \delta_2 \leq (2 - 1) = 1, \ 0 \leq \delta_3 \leq (4 - 2) = 2. \]

If $x = 1.5$ then $\delta_1 = 1, \ \delta_2 = 0.5, \ \delta_3 = 0$. Then

\[ g(x) = \sum_{p=1}^{\bar{p}} g(P_{p-1} + \delta_p) - \sum_{p=1}^{\bar{p}-1} g(P_p) = g(0 + \delta_1) + g(1 + \delta_2) + g(2 + \delta_3) - g(1) - g(2) \]
The Lower Bounding problem $Q$: step 2

$$0 \leq \delta_1 \leq (1 - 0) = 1,\ 0 \leq \delta_2 \leq (2 - 1) = 1,\ 0 \leq \delta_3 \leq (4 - 2) = 2.$$ If $x = 1.5$ then $\delta_1 = 1,\ \delta_2 = 0.5,\ \delta_3 = 0$. Then

$$g(x) = \sum_{p=1}^{\bar{p}} g(P_{p-1} + \delta_p) - \sum_{p=1}^{\bar{p}-1} g(P_p) = g(0 + \delta_1) + g(1 + \delta_2) + g(2 + \delta_3) - g(1) - g(2) = g(1) + g(1.5) + g(2) - g(1) - g(2)$$
The Lower Bounding problem $Q$: step 2

$0 \leq \delta_1 \leq (1 - 0) = 1, \ 0 \leq \delta_2 \leq (2 - 1) = 1, \ 0 \leq \delta_3 \leq (4 - 2) = 2.$

If $x = 1.5$ then $\delta_1 = 1, \ \delta_2 = 0.5, \ \delta_3 = 0$. Then

$$g(x) = \sum_{p=1}^{\overline{p}} g(P_{p-1} + \delta_p) - \sum_{p=1}^{\overline{p}-1} g(P_p) = g(0 + \delta_1) + g(1 + \delta_2) + g(2 + \delta_3) - g(1) - g(2) = g(1) + g(1.5) + g(2) - g(1) - g(2) = g(1.5)$$
Still nonconvex;
The Lower Bounding problem $Q$: step 3

Still nonconvex;

Use piece-wise linear approximation for the concave intervals:
The Lower Bounding problem $Q$: step 3

Still nonconvex;

Use piece-wise linear approximation for the concave intervals:
The Lower Bounding problem $Q$: step 3

Still nonconvex;

Use piece-wise linear approximation for the concave intervals:
The Lower Bounding problem $Q$: the convex MINLP model

Replace the term $g(x_k)$ with:

$$\sum_{p \in \bar{H}} g(P_{p-1} + \delta_p) + \sum_{p \in \hat{H}} \sum_{b \in B_p} g(X_{p,b}) \alpha_{p,b} - \sum_{p=1}^{\bar{P}-1} g(P_p),$$

and we include the following set of new constraints:

$$P_0 + \sum_{p=1}^{\bar{P}} \delta_p - x_k = 0;$$

$$\delta_p - (P_p - P_{p-1})z_p \geq 0, \forall p \in \bar{H} \cup \hat{H};$$

$$\delta_p - (P_p - P_{p-1})z_{p-1} \leq 0, \forall p \in \bar{H} \cup \hat{H};$$
The Lower Bounding problem $Q$: the convex MINLP model

Replace the term $g(x_k)$ with:

$$\sum_{p \in \hat{H}} g(P_{p-1} + \delta_p) + \sum_{p \in \hat{H}} \sum_{b \in B_p} g(X_{p,b}) \alpha_{p,b} - \sum_{p=1}^{\bar{P}-1} g(P_p),$$

and we include the following set of new constraints:

$$P_0 + \sum_{p=1}^{\bar{P}} \delta_p - x_k = 0;$$

$$\delta_p - (P_p - P_{p-1})z_p \geq 0, \forall p \in \hat{H} \cup \hat{H};$$

$$\delta_p - (P_p - P_{p-1})z_{p-1} \leq 0, \forall p \in \hat{H} \cup \hat{H};$$

$$P_{p-1} + \delta_p - \sum_{b \in B_p} X_{p,b} \alpha_{p,b} = 0, \forall p \in \hat{H};$$
The Lower Bounding problem $Q$: the convex MINLP model

Replace the term $g(x_k)$ with:

$$\sum_{p \in \check{H}} g(P_{p-1} + \delta_p) + \sum_{p \in \hat{H}} \sum_{b \in B_p} g(X_{p,b}) \alpha_{p,b} - \sum_{p=1}^{\bar{p}-1} g(P_{p}) ,$$

and we include the following set of new constraints:

$$P_0 + \sum_{p=1}^{\bar{p}} \delta_p - x_k = 0 ;$$

$$\delta_p - (P_p - P_{p-1})z_p \geq 0 , \forall p \in \check{H} \cup \hat{H} ;$$

$$\delta_p - (P_p - P_{p-1})z_{p-1} \leq 0 , \forall p \in \check{H} \cup \hat{H} ;$$

$$P_{p-1} + \delta_p - \sum_{b \in B_p} X_{p,b} \alpha_{p,b} = 0 , \forall p \in \hat{H} ;$$

$$\sum_{b \in B_p} \alpha_{p,b} = 1 , \forall p \in \hat{H} ;$$

$$\{\alpha_{p,b} : b \in B_p\} := \text{SOS2} , \forall p \in \hat{H} .$$

with two dummy variables $z_0 := 1$, $z_{\bar{p}} := 0$ and the new set of variables $\alpha_{p,b}.$
The Upper Bounding problem $\mathcal{R}$

Upper Bound of the original problem:
1. The integer variables are fixed;
The Upper Bounding problem \( \mathcal{R} \)

Upper Bound of the original problem:

1. The integer variables are fixed;
2. We solve the resulting non-convex NLP problem to local optimality;

\[
\begin{align*}
\min & \quad \sum_{j \in N} C_j x_j \\
\text{subject to} & \quad f(x) \leq 0 \\
& \quad r_i(x) + \sum_{k \in H(i)} g_{ik}(x_k) \leq 0, \quad \forall i \in M \\
& \quad L_j \leq x_j \leq U_j, \quad \forall j \in N \\
& \quad x_j = x_j, \quad \forall j \in I.
\end{align*}
\]

(\( \mathcal{R} \))
Choose tolerances $\varepsilon, \varepsilon_{\text{feas}} > 0$; initialize $LB := -\infty$; $UB := +\infty$;
Find $P_p^i, \hat{H}^i, \check{H}^i, X_{pb}^i (\forall i \in M, p \in \{1 \ldots p^i\}, b \in B_p^i)$. 
SC-MINLP (Sequential Convex MINLP) Algorithm

Choose tolerances $\varepsilon, \varepsilon_{\text{feas}} > 0$; initialize $LB := -\infty$; $UB := +\infty$;
Find $P^i_p, \hat{H}^i, \tilde{H}^i, X^i_{pb}$ ($\forall i \in M, p \in \{1 \ldots \bar{p}^i\}, b \in B^i_p$).

repeat
  Solve the convex MINLP relaxation $Q$ of the original problem $P$ to obtain $x$;
  if ($\text{val}(Q) > LB$) then
    $LB := \text{val}(Q)$;
    if ($x$ is feasible for the original problem $P$ (within tolerance $\varepsilon_{\text{feas}}$)) then
      return $x$
    end if
  end if
end if
Choose tolerances $\varepsilon, \varepsilon_{\text{feas}} > 0$; initialize $LB := -\infty$; $UB := +\infty$;
Find $P^i_p, \hat{H}^i, \check{H}^i, X^i_{pb}$ ($\forall i \in M, p \in \{1 \ldots p_i\}, b \in B^i_p$).

repeat
  Solve the convex MINLP relaxation $Q$ of the original problem $P$ to obtain $x$;
  if ($\text{val}(Q) > LB$) then
    $LB := \text{val}(Q)$;
    if ($x$ is feasible for the original problem $P$ (within tolerance $\varepsilon_{\text{feas}}$)) then
      return $x$
    end if
  end if
  Solve the non-convex NLP restriction $R$ of the original problem $P$ to obtain $\bar{x}$;
  if (solution $\bar{x}$ could be computed and $\text{val}(R) < UB$) then
    $UB := \text{val}(R)$; $x_{UB} := \bar{x}$
  end if
repeat
Choose tolerances $\varepsilon, \varepsilon_{\text{feas}} > 0$; initialize $LB := -\infty$; $UB := +\infty$; 
Find $P^i_p, \hat{H}^i, \check{H}^i, X^i_{pb} (\forall i \in M, p \in \{1 \ldots \overline{p}^i\}, b \in B^i_p)$.
repeat
  Solve the convex MINLP relaxation $Q$ of the original problem $P$ to obtain $\mathbf{x}$;
  if ($\text{val}(Q) > LB$) then
    $LB := \text{val}(Q)$;
    if ($\mathbf{x}$ is feasible for the original problem $P$ (within tolerance $\varepsilon_{\text{feas}}$)) then
      return $\mathbf{x}$
  end if
end if
Solve the non-convex NLP restriction $R$ of the original problem $P$ to obtain $\mathbf{x}$;
if (solution $\mathbf{x}$ could be computed and $\text{val}(R) < UB$) then
  $UB := \text{val}(R)$; $x_{UB} := \mathbf{x}$
end if
if ($UB - LB > \varepsilon$) then
  Update $B^i_p, X^i_{pb}$;
end if
Choose tolerances $\varepsilon, \varepsilon_{\text{feas}} > 0$; initialize $LB := -\infty$; $UB := +\infty$

Find $P^i_p, \hat{H}^i, \hat{H}^i, X^i_{pb} (\forall i \in M, p \in \{1 \ldots p^i\}, b \in B^i_p)$.

repeat

Solve the convex MINLP relaxation $Q$ of the original problem $P$ to obtain $x$;

if $(\text{val}(Q) > LB)$ then

$LB := \text{val}(Q)$;

if ($x$ is feasible for the original problem $P$ (within tolerance $\varepsilon_{\text{feas}}$)) then

return $x$

end if

end if

Solve the non-convex NLP restriction $R$ of the original problem $P$ to obtain $\bar{x}$;

if (solution $\bar{x}$ could be computed and $\text{val}(R) < UB$) then

$UB := \text{val}(R); x_{UB} := \bar{x}$

end if

if $(UB - LB > \varepsilon)$ then

Update $B^i_p, X^i_{pb}$

end if

until ($(UB - LB \leq \varepsilon)$ or (time or iteration limited exceeded))
Choose tolerances $\varepsilon, \varepsilon_{\text{feas}} > 0$; initialize $LB := -\infty$; $UB := +\infty$; Find $P_p^i$, $\hat{H}^i$, $\hat{\mathcal{H}}^i$, $X_{pb}^i$ ($\forall i \in M, p \in \{1 \ldots \bar{p}^i\}, b \in B_p^i$).

repeat
  Solve the convex MINLP relaxation $\mathcal{Q}$ of the original problem $\mathcal{P}$ to obtain $\mathbf{x}$;
  if $(\text{val}(\mathcal{Q}) > LB)$ then
    $LB := \text{val}(\mathcal{Q})$;
    if ($\mathbf{x}$ is feasible for the original problem $\mathcal{P}$ (within tolerance $\varepsilon_{\text{feas}}$)) then
      return $\mathbf{x}$
    end if
  end if
  Solve the non-convex NLP restriction $\mathcal{R}$ of the original problem $\mathcal{P}$ to obtain $\mathbf{x}$;
  if (solution $\mathbf{x}$ could be computed and $\text{val}(\mathcal{R}) < UB$) then
    $UB := \text{val}(\mathcal{R})$; $x_{UB} := \mathbf{x}$
  end if
  if ($UB - LB > \varepsilon$) then
    Update $B_p^i$, $X_{pb}^i$;
  end if
until ($(UB - LB \leq \varepsilon$) or (time or iteration limited exceeded))
return the current best solution $x_{UB}$
Add a breakpoint where the solution of problem $Q$ of the previous iteration lies (global convergence);
Refining the Lower Bounding problem

- Add a breakpoint where the solution of problem \( Q \) of the previous iteration lies (global convergence);
- Add a breakpoint where the solution of problem \( R \) of the previous iteration lies (speed up the convergence).
Convergence Analysis

Assumptions (\(l\) the iteration counter for the \texttt{repeat} loop):

A1. \(f(x)\) and \(r_i(x)\) continuous; \(g_{ik}\) in \((\mathcal{P})\) uniformly Lipschitz-continuous with a bounded Lipschitz constant \(L_g\).

A2. \(\mathcal{P}\) has a feasible point. For each \(l\), globally solve \(Q^l\).

A3. Adds a breakpoint for every \(Q\) problem solution \(x^l\).

A4. \(\varepsilon_{\text{feas}} = 0\), \(\varepsilon = 0\) and no time/iteration limit is set.

Theorem

Under assumptions A1-A4, Algorithm \texttt{SC-MINLP} either terminates at a global solution of the original problem \(\mathcal{P}\), or each limit point of the sequence \(\{x^l\}_{l=1}^\infty\) is a global solution of \(\mathcal{P}\).
Convergence Analysis

Assumptions (l the iteration counter for the repeat loop):

A1. $f(x)$ and $r_i(x)$ continuous; $g_{ik}$ in $(P)$ uniformly Lipschitz-continuous with a bounded Lipschitz constant $L_g$.

A2. $P$ has a feasible point. For each $l$, globally solve $Q^l$.

A3. Adds a breakpoint for every $Q$ problem solution $x^l$.

A4. $\varepsilon_{\text{feas}} = 0$, $\varepsilon = 0$ and no time/iteration limit is set.

Theorem

Under assumptions A1-A4, Algorithm SC-MINLP either terminates at a global solution of the original problem $P$, or each limit point of the sequence $\{x^l\}_{l=1}^{\infty}$ is a global solution of $P$.

The basic idea: at each iteration, the “error” of problem $Q$ is shrunk because of the first refinement rule.
Framework implemented under the AMPL environment;
Computational Results: Intro

- Framework implemented under the AMPL environment;
- Automatically detect the intervals of concavity/convexity using Matlab;
Computational Results: Intro

- Framework implemented under the AMPL environment;
- Automatically detect the intervals of concavity/convexity using Matlab;
- A solver for general convex MINLPs and a solver for NLPs used as black-box;
Framework implemented under the AMPL environment;
Automatically detect the intervals of concavity/convexity using Matlab;
A solver for general convex MINLPs and a solver for NLPs used as black-box;
Three classes of MINLPs of the form $\mathcal{P}$ (real-world applications);
Computational Results: Intro

- Framework implemented under the AMPL environment;
- Automatically detect the intervals of concavity/convexity using Matlab;
- A solver for general convex MINLPs and a solver for NLPs used as black-box;
- Three classes of MINLPs of the form $\mathcal{P}$ (real-world applications);
- Selected and reformulated instances of GlobalLib and MinlpLib.
Computational Results

Intel Core2 CPU 6600, 2.40 GHz, 1.94 GB of RAM. AMPL, Matlab R2007a, Ipopt, Bonmin. $\varepsilon_{\text{feas}} = 10^{-4}$, $\varepsilon = 10^{-5}$, time limit of 2 hours.
Computational Results

Intel Core2 CPU 6600, 2.40 GHz, 1.94 GB of RAM. AMPL, Matlab R2007a, Ipopt, Bonmin. $\varepsilon_{\text{feas}} = 10^{-4}$, $\varepsilon = 10^{-5}$, time limit of 2 hours.

<table>
<thead>
<tr>
<th>instance</th>
<th>var/int/cons</th>
<th>iter #</th>
<th>LB</th>
<th>UB</th>
<th>int change</th>
<th>time MINLP</th>
<th># br added</th>
</tr>
</thead>
</table>


Computational Results

Intel Core2 CPU 6600, 2.40 GHz, 1.94 GB of RAM. AMPL, Matlab R2007a, Ipopt, Bonmin. $\varepsilon_{\text{feas}} = 10^{-4}$, $\varepsilon = 10^{-5}$, time limit of 2 hours.

<table>
<thead>
<tr>
<th>instance</th>
<th>var/int/cons</th>
<th>iter #</th>
<th>LB</th>
<th>UB</th>
<th>int change</th>
<th>time MINLP</th>
<th># br added</th>
</tr>
</thead>
<tbody>
<tr>
<td>ufl.1</td>
<td>153/39/228</td>
<td>1</td>
<td>4,122.000</td>
<td>4,330.400</td>
<td>no</td>
<td>1.35</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2</td>
<td>4,324.780</td>
<td>4,330.400</td>
<td>no</td>
<td>11.84</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3</td>
<td>4,327.724</td>
<td>4,330.400</td>
<td>no</td>
<td>19.17</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4</td>
<td>4,328.993</td>
<td>4,330.400</td>
<td>no</td>
<td>30.75</td>
<td>5</td>
</tr>
<tr>
<td>205/65/254</td>
<td>5</td>
<td>4</td>
<td>4,330.070</td>
<td>4,330.400</td>
<td>no</td>
<td>45.42</td>
<td>5</td>
</tr>
</tbody>
</table>
Computational Results

Intel Core2 CPU 6600, 2.40 GHz, 1.94 GB of RAM. AMPL, Matlab R2007a, Ipopt, Bonmin. $\varepsilon_{\text{feas}} = 10^{-4}$, $\varepsilon = 10^{-5}$, time limit of 2 hours.

<table>
<thead>
<tr>
<th>instance</th>
<th>var/int/cons</th>
<th>iter #</th>
<th>LB</th>
<th>UB</th>
<th>int change</th>
<th>time MINLP</th>
<th># br added</th>
</tr>
</thead>
<tbody>
<tr>
<td>ufl_1</td>
<td>153/39/228</td>
<td>1</td>
<td>4,122.000</td>
<td>4,330.400</td>
<td>-</td>
<td>1.35</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2</td>
<td>4,324.780</td>
<td>4,330.400</td>
<td>no</td>
<td>11.84</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3</td>
<td>4,327.724</td>
<td>4,330.400</td>
<td>no</td>
<td>19.17</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4</td>
<td>4,328.993</td>
<td>4,330.400</td>
<td>no</td>
<td>30.75</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>205/65/254</td>
<td>5</td>
<td>4,330.070</td>
<td>4,330.400</td>
<td>no</td>
<td>45.42</td>
<td>5</td>
</tr>
<tr>
<td>ufl_2</td>
<td>189/57/264</td>
<td>1</td>
<td>27,516.600</td>
<td>27,516.569</td>
<td>-</td>
<td>4.47</td>
<td>-</td>
</tr>
</tbody>
</table>
Computational Results

Intel Core2 CPU 6600, 2.40 GHz, 1.94 GB of RAM. AMPL, Matlab R2007a, Ipopt, Bonmin. $\varepsilon_{\text{feas}} = 10^{-4}$, $\varepsilon = 10^{-5}$, time limit of 2 hours.

<table>
<thead>
<tr>
<th>instance</th>
<th>var/int/cons</th>
<th>iter #</th>
<th>LB</th>
<th>UB</th>
<th>int change</th>
<th>time MINLP</th>
<th># br added</th>
</tr>
</thead>
<tbody>
<tr>
<td>ufl_1</td>
<td>153/39/228</td>
<td>1</td>
<td>4,122.000</td>
<td>4,330.400</td>
<td>-</td>
<td>1.35</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>...</td>
<td>2</td>
<td>4,324.780</td>
<td>4,330.400</td>
<td>no</td>
<td>11.84</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>...</td>
<td>3</td>
<td>4,327.724</td>
<td>4,330.400</td>
<td>no</td>
<td>19.17</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>...</td>
<td>4</td>
<td>4,328.993</td>
<td>4,330.400</td>
<td>no</td>
<td>30.75</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>205/65/254</td>
<td>5</td>
<td>4,330.070</td>
<td>4,330.400</td>
<td>no</td>
<td>45.42</td>
<td>5</td>
</tr>
<tr>
<td>ufl_2</td>
<td>189/57/264</td>
<td>1</td>
<td>27,516.600</td>
<td>27,516.569</td>
<td>-</td>
<td>4.47</td>
<td>-</td>
</tr>
<tr>
<td>ufl_3</td>
<td>79/21/101</td>
<td>1</td>
<td>1,947.883</td>
<td>2,756.890</td>
<td>-</td>
<td>2.25</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>...</td>
<td>2</td>
<td>2,064.267</td>
<td>2,756.890</td>
<td>no</td>
<td>2.75</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>87/25/105</td>
<td>3</td>
<td>2,292.743</td>
<td>2,292.777</td>
<td>no</td>
<td>3.06</td>
<td>2</td>
</tr>
</tbody>
</table>
Computational Results

Intel Core2 CPU 6600, 2.40 GHz, 1.94 GB of RAM. AMPL, Matlab R2007a, Ipopt, Bonmin. $\varepsilon_{\text{feas}} = 10^{-4}$, $\varepsilon = 10^{-5}$, time limit of 2 hours.

<table>
<thead>
<tr>
<th>instance</th>
<th>var/int/cons</th>
<th>iter #</th>
<th>LB</th>
<th>UB</th>
<th>int change</th>
<th>time MINLP</th>
<th># br added</th>
</tr>
</thead>
<tbody>
<tr>
<td>ufl_1</td>
<td>153/39/228</td>
<td>1</td>
<td>4,122.000</td>
<td>4,330.400</td>
<td>-</td>
<td>1.35</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2</td>
<td>4,324.780</td>
<td>4,330.400</td>
<td>no</td>
<td>11.84</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3</td>
<td>4,327.724</td>
<td>4,330.400</td>
<td>no</td>
<td>19.17</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4</td>
<td>4,328.993</td>
<td>4,330.400</td>
<td>no</td>
<td>30.75</td>
<td>5</td>
</tr>
<tr>
<td>ufl_2</td>
<td>205/65/254</td>
<td>5</td>
<td>4,330.070</td>
<td>4,330.400</td>
<td>no</td>
<td>45.42</td>
<td>5</td>
</tr>
<tr>
<td>ufl_3</td>
<td>189/57/264</td>
<td>1</td>
<td>27,516.600</td>
<td>27,516.569</td>
<td>-</td>
<td>4.47</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>79/21/101</td>
<td>1</td>
<td>1,947.883</td>
<td>2,756.890</td>
<td>-</td>
<td>2.25</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2</td>
<td>2,064.267</td>
<td>2,756.890</td>
<td>no</td>
<td>2.75</td>
<td>2</td>
</tr>
<tr>
<td>hydro_1</td>
<td>87/25/105</td>
<td>3</td>
<td>2,292.743</td>
<td>2,292.777</td>
<td>no</td>
<td>3.06</td>
<td>2</td>
</tr>
<tr>
<td>hydro_2</td>
<td>324/142/445</td>
<td>1</td>
<td>-10,231.039</td>
<td>-10,140.763</td>
<td>-</td>
<td>18.02</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>332/146/449</td>
<td>2</td>
<td>-10,140.760</td>
<td>-10,140.763</td>
<td>no</td>
<td>23.62</td>
<td>4</td>
</tr>
<tr>
<td>hydro_3</td>
<td>324/142/445</td>
<td>1</td>
<td>-3,950.697</td>
<td>-3,891.224</td>
<td>-</td>
<td>21.73</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2</td>
<td>-3,950.583</td>
<td>-3,891.224</td>
<td>no</td>
<td>21.34</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3</td>
<td>-3,950.583</td>
<td>-3,891.224</td>
<td>no</td>
<td>27.86</td>
<td>2</td>
</tr>
<tr>
<td>hydro_3</td>
<td>336/148/451</td>
<td>4</td>
<td>-3,932.182</td>
<td>-3,932.182</td>
<td>no</td>
<td>38.20</td>
<td>2</td>
</tr>
<tr>
<td>hydro_3</td>
<td>324/142/445</td>
<td>1</td>
<td>-4,753.849</td>
<td>-4,634.409</td>
<td>-</td>
<td>59.33</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2</td>
<td>-4,719.927</td>
<td>-4,660.189</td>
<td>no</td>
<td>96.93</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3</td>
<td>-4,710.734</td>
<td>-4,710.734</td>
<td>yes</td>
<td>101.57</td>
<td>2</td>
</tr>
</tbody>
</table>
Computational Results

Intel Core2 CPU 6600, 2.40 GHz, 1.94 GB of RAM. AMPL, Matlab R2007a, Ipopt, Bonmin. $\varepsilon_{\text{feas}} = 10^{-4}, \varepsilon = 10^{-5}$, time limit of 2 hours.

<table>
<thead>
<tr>
<th>instance</th>
<th>var/int/cons</th>
<th>iter #</th>
<th>LB</th>
<th>UB</th>
<th>int change</th>
<th>time MINLP</th>
<th># br added</th>
</tr>
</thead>
<tbody>
<tr>
<td>nck_20_100</td>
<td>144/32/205</td>
<td>1</td>
<td>-162.444</td>
<td>-159.444</td>
<td>-</td>
<td>0.49</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>146/33/206</td>
<td>2</td>
<td>-159.444</td>
<td>-159.444</td>
<td>-</td>
<td>0.94</td>
<td>1</td>
</tr>
<tr>
<td>nck_20_200</td>
<td>144/32/205</td>
<td>1</td>
<td>-244.015</td>
<td>-238.053</td>
<td>-</td>
<td>0.67</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2</td>
<td>-241.805</td>
<td>-238.053</td>
<td>-</td>
<td>0.83</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3</td>
<td>-241.348</td>
<td>-238.053</td>
<td>-</td>
<td>1.16</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4</td>
<td>-240.518</td>
<td>-238.053</td>
<td>-</td>
<td>1.35</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5</td>
<td>-239.865</td>
<td>-238.053</td>
<td>-</td>
<td>1.56</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6</td>
<td>-239.744</td>
<td>-238.053</td>
<td>-</td>
<td>1.68</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>156/38/211</td>
<td>7</td>
<td>-239.125</td>
<td>-239.125</td>
<td>-</td>
<td>1.81</td>
<td>1</td>
</tr>
<tr>
<td>nck_20_450</td>
<td>144/32/205</td>
<td>1</td>
<td>-391.499</td>
<td>-391.337</td>
<td>-</td>
<td>0.79</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>146/32/206</td>
<td>2</td>
<td>-391.364</td>
<td>-391.337</td>
<td>-</td>
<td>0.87</td>
<td>1</td>
</tr>
<tr>
<td>nck_50_400</td>
<td>356/78/507</td>
<td>1</td>
<td>-518.121</td>
<td>-516.947</td>
<td>-</td>
<td>4.51</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2</td>
<td>-518.057</td>
<td>-516.947</td>
<td>-</td>
<td>14.94</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3</td>
<td>-517.837</td>
<td>-516.947</td>
<td>-</td>
<td>23.75</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4</td>
<td>-517.054</td>
<td>-516.947</td>
<td>-</td>
<td>25.07</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>372/86/515</td>
<td>5</td>
<td>-516.947</td>
<td>-516.947</td>
<td>-</td>
<td>31.73</td>
<td>2</td>
</tr>
<tr>
<td>nck_100_35</td>
<td>734/167/1035</td>
<td>1</td>
<td>-83.580</td>
<td>-79.060</td>
<td>-</td>
<td>3.72</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2</td>
<td>-82.126</td>
<td>-81.638</td>
<td>-</td>
<td>21.70</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3</td>
<td>-82.077</td>
<td>-81.638</td>
<td>-</td>
<td>6.45</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>744/172/1040</td>
<td>4</td>
<td>-81.638</td>
<td>-81.638</td>
<td>-</td>
<td>11.19</td>
<td>1</td>
</tr>
<tr>
<td>nck_100_80</td>
<td>734/167/1035</td>
<td>1</td>
<td>-174.841</td>
<td>-171.024</td>
<td>-</td>
<td>6.25</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2</td>
<td>-173.586</td>
<td>-172.631</td>
<td>-</td>
<td>24.71</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>742/171/1039</td>
<td>3</td>
<td>-172.632</td>
<td>-172.632</td>
<td>-</td>
<td>12.85</td>
<td>2</td>
</tr>
</tbody>
</table>
Comparison

<table>
<thead>
<tr>
<th>instance</th>
<th>var/int/cons original</th>
<th>SC-MINLP time (LB)</th>
<th>UB</th>
<th>COUENNE time (LB)</th>
<th>UB</th>
</tr>
</thead>
</table>


## Comparison

<table>
<thead>
<tr>
<th>instance</th>
<th>var/int/cons</th>
<th>SC-MINLP time (LB)</th>
<th>SC-MINLP UB</th>
<th>COUENNE time (LB)</th>
<th>COUENNE UB</th>
</tr>
</thead>
<tbody>
<tr>
<td>ufl_1</td>
<td>45/3/48</td>
<td>116.47</td>
<td>4,330.400</td>
<td>529.49</td>
<td>4,330.400</td>
</tr>
<tr>
<td>ufl_2</td>
<td>45/3/48</td>
<td>17.83</td>
<td>27,516.569</td>
<td>232.85</td>
<td>27,516.569</td>
</tr>
<tr>
<td>ufl_3</td>
<td>32/2/36</td>
<td>8.44</td>
<td>2,292.777</td>
<td>0.73</td>
<td>2,292.775</td>
</tr>
<tr>
<td>instance</td>
<td>var/int/cons</td>
<td>SC-MINLP</td>
<td>COUENNE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
<td>----------</td>
<td>---------</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>original</td>
<td>time (LB)</td>
<td>UB</td>
<td></td>
<td></td>
</tr>
<tr>
<td>uf1_1</td>
<td>45/3/48</td>
<td>116.47</td>
<td>4,330.400</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>529.49</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>4,330.400</td>
<td></td>
<td></td>
</tr>
<tr>
<td>uf1_2</td>
<td>45/3/48</td>
<td>17.83</td>
<td>27,516.569</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>232.85</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>27,516.569</td>
<td></td>
<td></td>
</tr>
<tr>
<td>uf1_3</td>
<td>32/2/36</td>
<td>8.44</td>
<td>2,292.777</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>0.73</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2,292.775</td>
<td></td>
<td></td>
</tr>
<tr>
<td>hydro_1</td>
<td>124/62/165</td>
<td>107.77</td>
<td>-10,140.763</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(-11,229.80)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>-10,140.763</td>
<td></td>
<td></td>
</tr>
<tr>
<td>hydro_2</td>
<td>124/62/165</td>
<td>211.79</td>
<td>-3,932.182</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(-12,104.40)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>-2,910.910</td>
<td></td>
<td></td>
</tr>
<tr>
<td>hydro_3</td>
<td>124/62/165</td>
<td>337.77</td>
<td>-4,710.734</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(-12,104.40)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>-3,703.070</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Comparison

<table>
<thead>
<tr>
<th>instance</th>
<th>var/int/cons</th>
<th>SC-MINLP</th>
<th>COUENNE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>original</td>
<td>time (LB)</td>
<td>time (LB)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>UB</td>
<td>UB</td>
</tr>
<tr>
<td>uf1_1</td>
<td>45/3/48</td>
<td>116.47</td>
<td>529.49</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4,330.400</td>
<td>4,330.400</td>
</tr>
<tr>
<td>uf1_2</td>
<td>45/3/48</td>
<td>17.83</td>
<td>232.85</td>
</tr>
<tr>
<td></td>
<td></td>
<td>27,516.569</td>
<td>27,516.569</td>
</tr>
<tr>
<td>uf1_3</td>
<td>32/2/36</td>
<td>8.44</td>
<td>0.73</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2,292.777</td>
<td>2,292.775</td>
</tr>
<tr>
<td>hydro_1</td>
<td>124/62/165</td>
<td>107.77</td>
<td>-10,140.763</td>
</tr>
<tr>
<td></td>
<td></td>
<td>-10,140.763</td>
<td>(-11,229.80)</td>
</tr>
<tr>
<td>hydro_2</td>
<td>124/62/165</td>
<td>211.79</td>
<td>-3,932.182</td>
</tr>
<tr>
<td></td>
<td></td>
<td>-3,932.182</td>
<td>(-12,104.40)</td>
</tr>
<tr>
<td>hydro_3</td>
<td>124/62/165</td>
<td>337.77</td>
<td>-4,710.734</td>
</tr>
<tr>
<td></td>
<td></td>
<td>-4,710.734</td>
<td>(-12,104.40)</td>
</tr>
<tr>
<td>nck_20_100</td>
<td>40/0/21</td>
<td>15.76</td>
<td>3.29</td>
</tr>
<tr>
<td></td>
<td></td>
<td>-159.444</td>
<td>-159.444</td>
</tr>
<tr>
<td>nck_20_200</td>
<td>40/0/21</td>
<td>23.76</td>
<td>(-352.86)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>-239.125</td>
<td>-238.053</td>
</tr>
<tr>
<td>nck_20_450</td>
<td>40/0/21</td>
<td>15.52</td>
<td>(-474.606)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>-391.337</td>
<td>-383.149</td>
</tr>
<tr>
<td>nck_50_400</td>
<td>100/0/51</td>
<td>134.25</td>
<td>(-1020.73)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>-516.947</td>
<td>-497.665</td>
</tr>
<tr>
<td>nck_100_35</td>
<td>200/0/101</td>
<td>110.25</td>
<td>90.32</td>
</tr>
<tr>
<td></td>
<td></td>
<td>-81.638</td>
<td>-81.638</td>
</tr>
<tr>
<td>nck_100_80</td>
<td>200/0/101</td>
<td>109.22</td>
<td>(-450.779)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>-172.632</td>
<td>-172.632</td>
</tr>
</tbody>
</table>
Thank you for your kind attention!