Linear Relaxations in Global Optimization: Gradient-based Method and Affine Reformulation Technique.

Jordan Ninin

LAB-STICC / ENSTA-Bretagne
Brest, France

May 8, 2013
Introduction

We consider global optimization of Mixed Integer Non Linear Programming problems in a deterministic and reliable way.

\[
\begin{align*}
\min_{x, y \in X \times Y \subset \mathbb{R}^n \times \mathbb{Z}^m} & \quad f(x, y) \\
\text{s.t.} & \quad g_l(x, y) \leq 0, \quad \forall l \in \{1, \ldots, p\}, \\
& \quad h_k(x, y) = 0, \quad \forall k \in \{1, \ldots, q\}.
\end{align*}
\]
We consider global optimization of Mixed Integer Non Linear Programming problems in a deterministic and reliable way.

\[
\begin{aligned}
\min_{x,y \in X \times Y \subset \mathbb{R}^n \times \mathbb{Z}^m} & \quad f(x, y) \\
\text{s.t.} & \quad g_l(x, y) \leq 0, \quad \forall l \in \{1, \ldots, p\}, \\
& \quad h_k(x, y) = 0, \quad \forall k \in \{1, \ldots, q\}.
\end{aligned}
\]

- Comparison and Combining different kinds of reliable linear relaxation method.

\[\Rightarrow\] Accelerate resolution of a Branch and Bound Algorithm based on Interval Analysis
Branch and Bound Algorithm based on Interval Analysis

Each iteration:

• Choice and Subdivision of the box $X$ into 2 boxes,
• Reduction of sub-boxes, $L$ ⇒ Constraint Propagations, Relaxation Techniques, ...
• Computation of lower bounds, $L$ ⇒ Interval Arithmetic, Relaxation Techniques, ...
• Elimination of boxes that cannot contain the global optimum, $L$ ⇒ Elts which do not satisfy constraints, lower bound $\tilde{f} > \min(Z, f_{z})$ in $L$

Else: Store in $L$

STOP $\Rightarrow \max(Z, f_{z}) \in L$

$\widetilde{Z} \leq \epsilon$

$L = \Rightarrow \tilde{f} - \min(Z, f_{z}) \in L$ $f_{z} \leq \epsilon$
Branch and Bound Algorithm based on Interval Analysis

Each iteration:

- **Choice and Subdivision of the box** $X$ (into 2 boxes),
  $\implies \mathcal{L}$ list of possible solutions
Branch and Bound Algorithm based on Interval Analysis

Each iteration:

- **Choice and Subdivision of the box** $\mathbf{X}$ (into 2 boxes),
  $\implies \mathcal{L}$ list of possible solutions
- **Reduction of sub-boxes,**
  $\implies$ Constraint Propagations, Relaxation Techniques, ...

$\mathcal{L} = \{ z \in \mathbb{R}^n \mid f(z) \leq \epsilon \}$
Branch and Bound Algorithm based on Interval Analysis

Each iteration:

- **Choice and Subdivision of the box** $\mathbf{X}$ (into 2 boxes),
  $\implies \mathcal{L}$ list of possible solutions

- **Reduction of sub-boxes,**
  $\implies$ Constraint Propagations, Relaxation Techniques, ...

- **Computation of lower bounds**
  $\implies$ Interval Arithmetic, Relaxation Techniques,...
Branch and Bound Algorithm based on Interval Analysis

Each iteration:

- **Choice and Subdivision of the box** \( X \) (into 2 boxes),
  \[ \Rightarrow \mathcal{L} \text{ list of possible solutions} \]
- **Reduction of sub-boxes**, 
  \[ \Rightarrow \text{Constraint Propagations, Relaxation Techniques, ...} \]
- **Computation of lower bounds**
  \[ \Rightarrow \text{Interval Arithmetic, Relaxation Techniques,...} \]
- **Elimination** of boxes that cannot contain the global optimum
  \[ \Rightarrow \text{Elts which do not satisfy constraints, lower bound } > \tilde{f},.. \]
Else: **Store in** \( \mathcal{L} \)
Branch and Bound Algorithm based on Interval Analysis

Each iteration:

- **Choice and Subdivision of the box** $\mathbf{X}$ (into 2 boxes),
  $\Rightarrow \mathcal{L}$ list of possible solutions

- **Reduction of sub-boxes,**
  $\Rightarrow$ Constraint Propagations, Relaxation Techniques, ...

- **Computation of lower bounds**
  $\Rightarrow$ Interval Arithmetic, Relaxation Techniques, ...

- **Elimination** of boxes that cannot contain the global optimum
  $\Rightarrow$ Elts which do not satisfy constraints, lower bound $\geq \tilde{f}$,

Else: **Store in** $\mathcal{L}$

- **STOP**
  $\Rightarrow \max_{(\mathbf{Z}, f_z) \in \mathcal{L}} \text{wid}(\mathbf{Z}) \leq \epsilon_L$
  $\Rightarrow \tilde{f} - \min_{(\mathbf{Z}, f_z) \in \mathcal{L}} f_z \leq \epsilon_f$
Branch and Bound Algorithm based on Interval Analysis

Each iteration:

- **Choice** and **Subdivision of the box** $\mathbf{X}$ (into 2 boxes),
  $\Rightarrow \mathcal{L}$ list of possible solutions

- **Reduction of sub-boxes,**
  $\Rightarrow$ Constraint Propagations, **Relaxation Techniques,** ...

- **Computation of lower bounds**
  $\Rightarrow$ Interval Arithmetic, **Relaxation Techniques,** ...

- **Elimination** of boxes that cannot contain the global optimum
  $\Rightarrow$ Elts which do not satisfy constraints, lower bound $> \tilde{f}$,..
  Else: Store in $\mathcal{L}$

- **STOP**
  $\Rightarrow \max_{(\mathbf{z}, f_\mathbf{z}) \in \mathcal{L}} \text{wid}(\mathbf{Z}) \leq \epsilon_L$
  $\Rightarrow \tilde{f} - \min_{(\mathbf{z}, f_\mathbf{z}) \in \mathcal{L}} f_\mathbf{z} \leq \epsilon_f$
Contents

1 Reformulation Method
   Principle
   Gradient-based Method
   Affine Reformulation Technique
   Reformulation-Linearization-Techniques

2 Numerical Results
1 Reformulation Method
   Principle
   Gradient-based Method
   Affine Reformulation Technique
   Reformulation-Linearization-Techniques

2 Numerical Results
Linear Relaxation Techniques

\[
\begin{align*}
\min_{x \in [x]} & \quad f(x) \\
\text{s.t.} & \quad g_i(x) \leq 0, \quad h_k(x) = 0.
\end{align*}
\]  

\[
\Rightarrow \quad \begin{align*}
\min_{y \in [y]} & \quad c^T y, \\
\text{s.t.} & \quad A y \leq b.
\end{align*}
\]  

\[
\forall (y, z) \in [y] \times [lb, best_{sol}],
\begin{align*}
z &= c^T y, \\
A y &\leq b.
\end{align*}
\]
Linear Relaxation Techniques

\[ \begin{cases} \min_{x \in [x]} f(x) \\
\text{s.t. } g_l(x) \leq 0, \\
\quad h_k(x) = 0. \end{cases} \Rightarrow \begin{cases} \min_{y \in [y]} c^T y, \\
\text{s.t. } A y \leq b. \end{cases} \Rightarrow \forall (y, z) \in [y] \times [lb, best\_sol], \\
\begin{cases} z = c^T y, \\
A y \leq b. \end{cases} \]

\( C([x] \times [lb, best\_sol]) \rightarrow \)

- Contract the hypercube with the polyhedral feasibility domain,
- Contract the gap between the best known solution and the lower bound.
Linear Relaxation Techniques

\[
\begin{align*}
\min_{x \in [x]} & \quad f(x) \\
\text{s.t.} & \quad g_i(x) \leq 0, \\
& \quad h_k(x) = 0.
\end{align*}
\]

\[
\Rightarrow \left\{ \begin{array}{l}
\min_{y \in [y]} & \quad c^T y, \\
\text{s.t.} & \quad A y \leq b.
\end{array} \right.
\]

\[
\Rightarrow \forall (y, z) \in [y] \times [lb, best\_sol],
\]

\[
\begin{align*}
\quad & \quad z = c^T y, \\
\quad & \quad A y \leq b.
\end{align*}
\]

\[
C([x] \times [lb, best\_sol]) \rightarrow
\]

- Contract the hypercube with the polyhedral feasibility domain,
- Contract the gap between the best known solution and the lower bound.

Certification the result of the LP solver
**Linear Relaxation Techniques**

\[
\begin{array}{l}
\min_{x \in [x]} \quad f(x) \\
\text{s.t.} \quad g_l(x) \leq 0, \\
\quad h_k(x) = 0.
\end{array}
\]

\[
\Rightarrow \quad \left\{ \begin{array}{l}
\min_{y \in [y]} \quad c^T y, \\
\text{s.t.} \quad A y \leq b.
\end{array} \right.
\]

\[
\Rightarrow \quad \forall (y, z) \in [y] \times [lb, best\_sol],
\]

\[
\begin{array}{l}
z = c^T y, \\
A y \leq b.
\end{array}
\]

\[
C([x] \times [lb, best\_sol]) \rightarrow
\]

- Contract the hypercube with the polyhedral feasibility domain,
- Contract the gap between the best known solution and the lower bound.

**Certification the result of the LP solver**

- Linear Interval Program \(\Rightarrow\) linear interval solver (LURUPA)
Linear Relaxation Techniques

\[
\begin{align*}
\min_{x \in [x]} & \quad f(x) \\
\text{s.t.} & \quad g_i(x) \leq 0, \\
& \quad h_k(x) = 0.
\end{align*}
\]

\[\Rightarrow\]

\[
\begin{align*}
\min_{y \in [y]} & \quad c^T y, \\
\text{s.t.} & \quad Ay \leq b.
\end{align*}
\]

\[\Rightarrow\]

\[\forall (y, z) \in [y] \times [lb, best\_sol], \quad z = c^T y, \\
\text{such that } Ay \leq b.\]

\[C([x] \times [lb, best\_sol]) \rightarrow\]

- Contract the hypercube with the polyhedral feasibility domain,
- Contract the gap between the best known solution and the lower bound.

Certification the result of the LP solver

- Linear Interval Program \(\Rightarrow\) linear interval solver (LURUPA)
- Reliable Linear Program \(\Rightarrow\) Computing the residual of the dual by Interval Arithmetic

1 Reformulation Method
   Principle
   Gradient-based Method
   Affine Reformulation Technique
   Reformulation-Linearization-Techniques

2 Numerical Results
Inclusion Functions based on Taylor’s Expansions

Let \( f \) be a univariate differentiable function, and \( x, y \) and \( \xi \), 3 variables of \( X \) an interval of \( \mathbb{R} \).

\[
f(x) = f(y) + (x - y)f'(y) + \frac{(x - y)^2}{2} f''(y) + \ldots + \frac{(x - y)^n}{n!} f^{(n)}(\xi)
\]
Gradient-based Method

Inclusion Functions based on Taylor’s Expansions

Let $f$ be a univariate differentiable function, and $x$, $y$ and $\xi$, 3 variables of $X$ an interval of $\mathbb{R}$.

$$f(x) = f(y) + (x - y)f'(y) + \frac{(x - y)^2}{2} f''(y) + \ldots + \frac{(x - y)^n}{n!} f^{(n)}(\xi)$$

Let denote $F^{(n)}(X)$ an enclosure of $f^{(n)}(\xi)$ over $X$ (computed with an interval automatic differentiation tool).

Hence,

$$f(x) \in f(y) + (x - y)f'(y) + \frac{(x - y)^2}{2} f''(y) + \ldots + \frac{(x - y)^n}{n!} F^{(n)}(X), \forall y \in X, n \geq 0$$
Gradient-based Method

Inclusion Functions based on Taylor’s Expansions

Let $f$ be a \textit{univariate} differentiable function, and $x$, $y$ and $\xi$, 3 variables of $X$ an interval of $\mathbb{R}$.

\[
f(x) = f(y) + (x - y)f'(y) + \frac{(x - y)^2}{2} f''(y) + \ldots + \frac{(x - y)^n}{n!} f^{(n)}(\xi)
\]

Let denote $F^{(n)}(X)$ an enclosure of $f^{(n)}(\xi)$ over $X$ (computed with an interval automatic differentiation tool).

Hence,

\[
f(x) \in f(y) + (x - y)f'(y) + \frac{(x - y)^2}{2} f''(y) + \ldots + \frac{(x - y)^n}{n!} F^{(n)}(X), \forall y \in X,
\]

\[\implies\text{ Inclusion functions:}\]

\[
T_1(y, X) = f(y) + (X - y)F'(X)
\]
Representation of the Taylor Inclusion Function

\[ f(m([x])) + \overline{f'}(x-m([x])) \]

\[ f(x) + \overline{f'}(x-x) \]
Gradient-based Method

X-Newton Method: I. Araya, G. Trombettonni, B. Neveu

- Choose several points among the $2^n$ corners of the hypercube:
  $\Rightarrow$ Different heuristics could be used.
Choose several points among the $2^n$ corners of the hypercube:

- Different heuristics could be used.
- Compute the linear relaxation associated to each chosen corner.
X-Newton Method: I. Araya, G. Trombettoni, B. Neveu

- Choose several points among the $2^n$ corners of the hypercube:
  $\Rightarrow$ Different heuristics could be used.
- Compute the linear relaxation associated to each chosen corner.
- Generate the linear program and Solve it.
Choose several points among the $2^n$ corner of the hypercube: 
⇒ Different heuristics could be used.

Compute the linear relaxation associated to each chosen corner.

Generate the linear program and Solve it.

Validate the result with the Neumaier-Shcherbina’s criteria.
Affine Reformulation Technique

Contents

1 Reformulation Method
   Principle
   Gradient-based Method
   Affine Reformulation Technique
   Reformulation-Linearization-Techniques

2 Numerical Results
Affine Reformulation Technique


Definition

Each quantity is represented by an affine form \( \hat{x} \)

\[
\hat{x} = x_0 + \sum_{i=1}^{n} x_i \epsilon_i ,
\]

with \( \forall i \in [0, n], x_i \in \mathbb{R} \) and \( \epsilon_i = [-1, 1] \).

Definition

Each quantity is represented by an affine form \( \hat{x} \)

\[
\hat{x} = x_0 + \sum_{i=1}^{n} x_i \epsilon_i,
\]

with \( \forall i \in [0, n], x_i \in \mathbb{R} \) and \( \epsilon_i = [-1, 1] \).

- Example: \( A = [1, 3] \) and \( B = [-2, 0] \),

\[
\begin{align*}
\hat{A} & \rightarrow 2 + \epsilon_1, \\
\hat{B} & \rightarrow -1 + \epsilon_2,
\end{align*}
\]
Affine Reformulation Technique


Definition

*Each quantity is represented by an affine form* \( \hat{x} \)

\[
\hat{x} = x_0 + \sum_{i=1}^{n} x_i \epsilon_i,
\]

*with* \( \forall i \in [0, n], x_i \in \mathbb{R} \) *and* \( \epsilon_i = [-1, 1] \).

- **Example:** \( A = [1, 3] \) and \( B = [-2, 0] \),

  \[
  \hat{A} \rightarrow 2 + \epsilon_1, \\
  \hat{B} \rightarrow -1 + \epsilon_2, \\
  1 + \hat{A} = 3 + \epsilon_1, \\
  5 \times \hat{B} = -5 + 5\epsilon_2, \\
  \hat{A} + \hat{B} = 1 + \epsilon_1 + \epsilon_2.
  \]
Non-Affine Operator

### Multiplication

\[
\hat{x} \times \hat{y} = \left( x_0 + \sum_{i=1}^{n+1} x_i \epsilon_i \right) \times \left( y_0 + \sum_{i=1}^{n+1} y_i \epsilon_i \right),
\]

\[
= x_0 y_0 + \sum_{i=1}^{n} \left( x_0 y_i + x_i y_0 \right) \epsilon_i + \left( x_0 y_{n+1} + x_{n+1} y_0 + \left( \sum_{i=1}^{n+1} |x_i| \times \sum_{i=1}^{n+1} |y_i| \right) \right) \epsilon_{\pm}.
\]

### Log, exp, \( \sqrt{\cdot} \), power, cos, ...

\[
\hat{f}(\hat{x}) = \zeta + \alpha \hat{x} + \delta \epsilon_{\pm},
\]

with \( \alpha, \delta, \zeta \in \mathbb{R} \) and \( \hat{x} = x_0 + \sum_{i=1}^{n} x_i \epsilon_i \).
Visualization of AA by expression tree

∀x ∈ [1, 2] × [2, 6], f(x) = x_1 x_2^2 − \exp(x_1 + x_2)
Affine Reformulation Technique

Visualization of AA by expression tree

∀x ∈ [1, 2] × [2, 6], f(x) = x₁x² − exp(x₁ + x₂)
Visualization of AA by expression tree

\( \forall x \in [1, 2] \times [2, 6], f(x) = x_1 x_2 - \exp(x_1 + x_2) \)
Visualization of AA by expression tree

\[ \forall x \in [1, 2] \times [2, 6], f(x) = x_1 x_2^2 - \exp(x_1 + x_2) \]
Visualization of AA by expression tree

∀x ∈ \([1, 2] \times [2, 6]\), \(f(x) = x_1x_2 - \exp(x_1 + x_2)\)
Visualization of AA by expression tree

\[ \forall x \in [1, 2] \times [2, 6], f(x) = x_1 x_2^2 - \exp(x_1 + x_2) \]
Visualization of AA by expression tree

∀x ∈ [1, 2] × [2, 6], f(x) = x_1 x_2^2 − \exp(x_1 + x_2)

\[ -1476.52 - 2.04\epsilon_1 - 16.17\epsilon_2 + 14446.22\epsilon_\pm \]

\[ 24 + 8\epsilon_1 + 24\epsilon_2 + 16\epsilon_\pm \]

\[ 16 + 16\epsilon_2 + 4\epsilon_\pm \]

\[ 1.5 + 0.5\epsilon_1 \]

\[ 4 + 2\epsilon_2 \]

\[ 1.5 + 0.5\epsilon_1 \]

\[ 4 + 2\epsilon_2 \]

\[ 1500.52 + 10.04\epsilon_1 + 40.17\epsilon_2 + 1430.22\epsilon_\pm \]

\[ 5.5 + 0.5\epsilon_1 + 2\epsilon_2 \]
Affine Reformulation Technique

Visualization of AA by expression tree

∀x ∈ [1, 2] × [2, 6], f(x) = x_1 x_2 − \exp(x_1 + x_2) ∈ [−2940.9579, −12.0855]

\[-1476.52 - 2.04\epsilon_1 - 16.17\epsilon_2 + 14446.22\epsilon_±\]

24 + 8\epsilon_1 + 24\epsilon_2 + 16\epsilon_±

1500.52 + 10.04\epsilon_1 + 40.17\epsilon_2 + 1430.22\epsilon_±

1.5 + 0.5\epsilon_1

16 + 16\epsilon_2 + 4\epsilon_±

4 + 2\epsilon_2

1.5 + 0.5\epsilon_1

5.5 + 0.5\epsilon_1 + 2\epsilon_2

4 + 2\epsilon_2

1.5 + 0.5\epsilon_1

4 + 2\epsilon_2
Graphical Representation

- Affine form
- Error of affine form
- Interval under study
- Lower bound

\[ f(x) \]
Affine Reformulation Technique

AF1 and AF2 $\Rightarrow$ automated way to linearize every function.
Affine Reformulation Technique

AF1 and AF2 ⇒ automated way to linearize every function.

\[ n \text{ fixed} \Rightarrow \text{affine transformation } T \text{ between} \]
\[ x \in X \subset \mathbb{R}^n \text{ and } z \in \varepsilon = [-1, 1]^n. \]
Affine Reformulation Technique

AF1 and AF2 ⇒ automated way to linearize every function.

\[ n \text{ fixed} \Rightarrow \text{affine transformation } \mathcal{T} \text{ between } \]
\[ x \in X \subset \mathbb{R}^n \text{ and } z \in \epsilon = [-1, 1]^n. \]

\[ \hat{f}(x) = f_0 + \sum_{i=1}^{n} f_i \epsilon_i + f_\pm \epsilon_\pm. \]
Affine Reformulation Technique

Affine Reformulation Technique: J. Ninin, F. Messine, P. Hansen

AF1 and AF2 ⇒ automated way to linearize every function.

\[ n \text{ fixed} \Rightarrow \text{affine transformation } T \text{ between } \]
\[ x \in X \subset \mathbb{R}^n \text{ and } z \in \epsilon = [-1, 1]^n. \]

\[
\hat{f}(x) = f_0 + \sum_{i=1}^{n} f_i z_i + f_\pm \epsilon_\pm. 
\]

Linear Approximation
**Affine Reformulation Technique**

**Affine Reformulation Technique: J.Ninin, F.Messine, P.Hansen**

AF1 and AF2 ⇒ automated way to linearize every function.

\( n \text{ fixed} \Rightarrow \text{affine transformation } T \text{ between } x \in X \subset \mathbb{R}^n \text{ and } z \in \epsilon = [-1, 1]^n. \)

\[
\hat{f}(x) = f_0 + \sum_{i=1}^{n} f_i z_i + f_\pm \epsilon_\pm.
\]

Linear Approximation
Affine Reformulation Technique

**Affine Reformulation Technique: J. Ninin, F. Messine, P. Hansen**

**AF1 and AF2** ⇒ automated way to linearize every function.

$n \text{ fixed} \Rightarrow \text{affine transformation } T \text{ between } x \in X \subset \mathbb{R}^n \text{ and } z \in \epsilon = [-1, 1]^n.$

\[
\hat{f}(x) = f_0 + \sum_{i=1}^{n} f_i z_i + f_{\pm} \epsilon_{\pm}.
\]

Linear Approximation

\[
\forall x \in X, z = T(x), f(x) - \sum_{i=1}^{n} f_i z_i \in [f_0 - f_{\pm}, f_0 + f_{\pm}]
\]
Reformulation Method

Affine Reformulation Technique

Reformulation of a NLP problem

\[
\begin{aligned}
\min_{x \in X} & \quad f(x) \\
\text{s. t.} & \quad g_l(x) \leq 0, \quad \forall l \in \{1, \ldots, p\}, \\
& \quad h_k(x) = 0, \quad \forall k \in \{1, \ldots, q\}.
\end{aligned}
\]
Reformulation of a NLP problem

\[
\begin{align*}
\min_{x \in \mathcal{X}} & \quad f(x) \\
\text{s. t.} & \quad g_l(x) \leq 0, \quad \forall l \in \{1, \ldots, p\}, \\
& \quad h_k(x) = 0, \quad \forall k \in \{1, \ldots, q\}.
\end{align*}
\]

⇒ Reformulate each equation with Affine Arithmetic ⇒
Reformulation of a NLP problem

\[
\begin{align*}
\min_{x \in X} & \quad f(x) \\
\text{s. t.} & \quad g_l(x) \leq 0, \quad \forall l \in \{1, \ldots, p\}, \\
& \quad h_k(x) = 0, \quad \forall k \in \{1, \ldots, q\}.
\end{align*}
\]

⇒ Reformulate each equation with Affine Arithmetic ⇒

\[
\begin{align*}
\min_{z \in [-1,1]^n} & \quad \sum_{i=1}^{n} f_i z_i \\
\text{s. t.} & \quad \sum_{i=1}^{n} (g_l)_i z_i \leq (g_l)_0 \pm (g_l)_0, \quad \forall l \in \{1, \ldots, p\}, \\
& \quad \sum_{i=1}^{n} (h_k)_i z_i \leq (h_k)_0 \pm (h_k)_0, \quad \forall k \in \{1, \ldots, q\}, \\
& \quad -\sum_{i=1}^{n} (h_k)_i z_i \leq (h_k)_0 \pm (h_k)_0, \quad \forall k \in \{1, \ldots, q\}.
\end{align*}
\]
Contents

1 Reformulation Method
   Principle
   Gradient-based Method
   Affine Reformulation Technique

2 Numerical Results
Reformulation-Linearization-Techniques

\[
\begin{align*}
\min_{x,y} & \quad f(x, y) \\
\text{s.t.} & \quad g_i(x, y) \leq 0, \\
& \quad h_k(x, y) = 0.
\end{align*}
\]
Reformulation-Linearization-Techniques

\[
\begin{align*}
\min_{x,y} & \quad f(x, y) \\
\text{s.t.} & \quad g_l(x, y) \leq 0, \\
& \quad h_k(x, y) = 0.
\end{align*}
\]

\[
\Rightarrow
\begin{align*}
\min_{x,y,w} & \quad w_{k0} \\
\text{s.t.} & \quad w_{k1} = x_1 y_1, \\
& \quad w_{k2} = \exp(x_5), \\
& \quad w_{k3} = w_{k1} w_{k2}, \\
& \quad w_{k3} = y_4 / w_{k3}, \\
& \quad \vdots
\end{align*}
\]

\[
\Rightarrow
\begin{align*}
\min_{x,y,w} & \quad w_{k0} \\
\text{linear relaxation}
\end{align*}
\]


Contents

1 Reformulation Method
   Principle
   Gradient-based Method
   Affine Reformulation Technique
   Reformulation-Linearization-Techniques

2 Numerical Results
Integration in IBEX: G.Chabert et al.

IBEX is a library containing a deterministic global optimization algorithm based on Interval Arithmetic.

- Compare XNewton reformulation, ART and a combination,
- Improve only the lower bound or Contract the domain of each variable.
IBEX is a library containing a deterministic global optimization algorithm based on Interval Arithmetic.

- Compare XNewton reformulation, ART and a combination,
- Improve only the lower bound or Contract the domain of each variable.

161 problems from the COCONUT database
(a library of global optimization test problems)

less than 50 variables
## Comparison: Contracting the box

<table>
<thead>
<tr>
<th>Method</th>
<th>Nb of success</th>
<th>Nb Success only by</th>
<th>Time</th>
<th>Time only if success</th>
</tr>
</thead>
<tbody>
<tr>
<td>ART</td>
<td>128</td>
<td>3</td>
<td>140.67 s</td>
<td>24.44 s</td>
</tr>
<tr>
<td>XNewton</td>
<td>128</td>
<td>3</td>
<td>143.06 s</td>
<td>28.35 s</td>
</tr>
<tr>
<td>Combining</td>
<td>131</td>
<td>-</td>
<td>132.03 s</td>
<td>29.16 s</td>
</tr>
</tbody>
</table>
Performance Profiles

![Graph showing performance profiles for different methods]

- Xnewton
- ART
- Combi
Conclusion

Preliminary results:

- Gradient-based Method and Affine Arithmetic-based method seem to be equivalent.
- The combination slows down the performance, but we need to test the merge of the two linearizations into one LP.

IBEX
http://www.emn.fr/z-info/ibex/