Constraint Techniques for a Safe and Fast Implementation of Optimality-Based Reduction

Yahia Lebbah, *Claude Michel* and Michel Rueher

I3S (UNSA-CNRS)

Global optimization

- We consider here the global optimization problem $\mathcal{P}$

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & g_i(x) = 0,\ i = 1..k \\ & g_j(x) \leq 0,\ j = k+1..m \end{array} \qquad (1)$$

with

  - $x \in \mathbf{x} \subset \mathbb{R}^n$, $f : \mathbb{R}^n \rightarrow \mathbb{R}$ and $g_j : \mathbb{R}^n \rightarrow \mathbb{R}$;
  - functions f and g_j are continuously differentiable on $\mathbf{x}$

Branch and bound

- Aim: *rigorously* solving $\mathcal{P}$

  $\longrightarrow$ find an *interval* $[L, U] \ni min f(x)$ such that $U - L \leq \epsilon$

  where U, L and ϵ are *floating point numbers*.

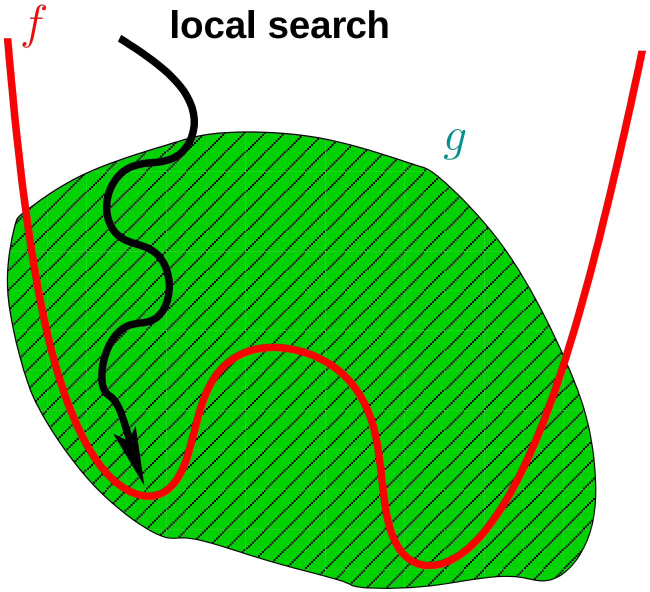
- Means : *branch and bound* algorithm
  - Based on a *reduction* approach which attempt to
    1. *reduce* the value of U (upper bounding),
    2. *increase* the value of L (lower bounding).
  - With the help of a *pruning* step.

Branch and bound algorithm

```
Function BB(IN $\mathbf{x}$, $\epsilon$; OUT $\mathcal{S}$, $[L,U]$)
$\mathcal{L} \leftarrow \{\mathbf{x}\}$;   $\mathcal{S} \leftarrow \emptyset$;   $(L,U) \leftarrow (-\infty,+\infty)$;
while $w([L,U]) > \epsilon$ do
    $\mathbf{x}' \leftarrow \mathbf{x}''$ such that $\underline{\mathbf{f}}_{\mathbf{x}''} = min\{\underline{\mathbf{f}}_{\mathbf{x}''} : \mathbf{x}'' \in \mathcal{L}\}$;   $\mathcal{L} \leftarrow \mathcal{L} \backslash \mathbf{x}'$;
    $\overline{\mathbf{f}}_{\mathbf{x}'} \leftarrow min(\overline{\mathbf{f}}_{\mathbf{x}'}, U)$;   $\mathbf{x}' \leftarrow Prune(\mathbf{x}')$;   $\underline{\mathbf{f}}_{\mathbf{x}'} \leftarrow LowerBound(\mathbf{x}')$;
    $(\overline{\mathbf{f}}_{\mathbf{x}'}, \mathbf{x}_p, Proved) \leftarrow UpperBox(\mathbf{x}')$;
    if $Proved$ then $\mathcal{S} \leftarrow \mathcal{S} \cup \{\mathbf{x}_p\}$; endif
    if $\mathbf{x}' \neq \emptyset$ then $(\mathbf{x}'_1, \mathbf{x}'_2) \leftarrow Split(\mathbf{x}')$;   $\mathcal{L} \leftarrow \mathcal{L} \cup \{\mathbf{x}'_1, \mathbf{x}'_2\}$;   endif
    if $\mathcal{L} = \emptyset$ then $(L,U) \leftarrow (+\infty,-\infty)$;
    else $(L,U) \leftarrow (min\{\underline{\mathbf{f}}_{\mathbf{x}''} : \mathbf{x}'' \in \mathcal{L}\}, min\{\overline{\mathbf{f}}_{\mathbf{x}''} : \mathbf{x}'' \in \mathcal{S}\})$;
    endif
endwhile
```

Upper bounding

- Aim: reduce the value of the global U
- Means: compute a feasible point !
  - local search
    $\rightarrow$ approximate feasible point x_{approx}
  - epsilon inflation process and proof
    $\rightarrow$ provide a feasible box x_{proved}
  - compute $\overline{f}^* = min(\overline{f}(x_{proved}), \overline{f}^*)$

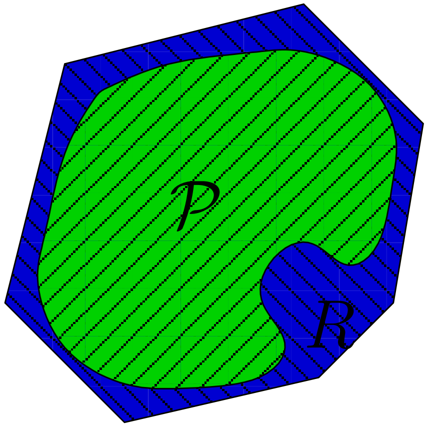


- Then, the *global* U is the smallest of the local $\overline{f}^*$

Lower bounding

- Aim: increase the value of the global L
- Means: relaxing the problem
  - linear relaxation R of $\mathcal{P}$

$$\begin{aligned} & min \quad d^T x \\ & s.t. \quad Ax \leq b \end{aligned} \tag{2}$$

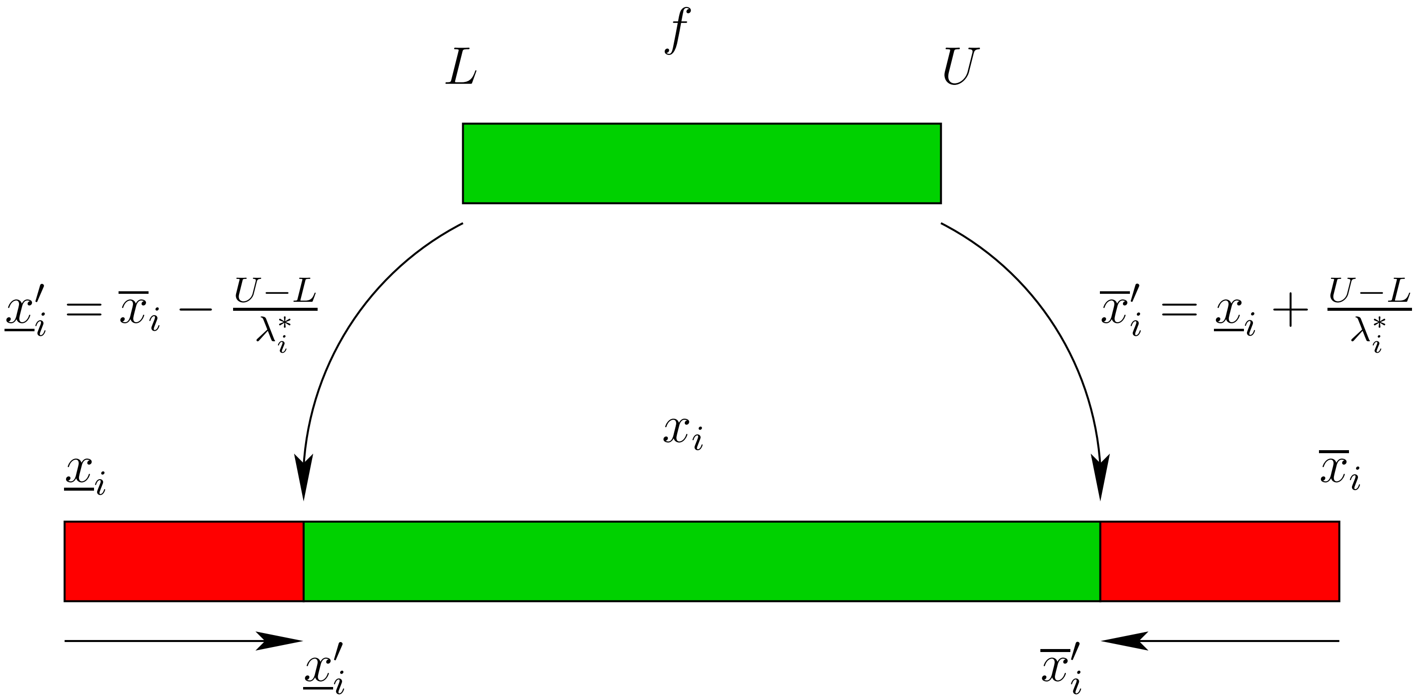


  - LP solver $\rightarrow \underline{f}^*$
- Issues:
  - efficient LP solvers work with floats
    $\rightarrow$ compute a *safe lower bound* of the min
  - relaxation coefficients are computed
    $\rightarrow$ must be *rigorously* computed

Then, the global L is the smallest of the local $\underline{f}^*$

Optimality Base Reduction (OBR)

- Why ? there is room for improvement ...
- OBR is a way to speed up the reduction process
- How ? (Ryoo & Sahinidis 96)



- does not modify the very branch and bound process

Theorems of OBR

- The theorem says that knowing $[L, U]$ (the domain of f), if the constraint $x_i - \overline{x}_i \leq 0$ is active at the optimal solution of R (i.e. $x_i - \overline{x}_i = 0$ if x_i is set to this optima) and has a corresponding multiplier $\lambda_i^* > 0$ (i.e. λ^* is the optimal solution of the dual of R). Then

$$x_i \geq \underline{x}'_i \text{ with } \underline{x}'_i = \overline{x}_i - \frac{U - L}{\lambda_i^*}. \tag{3}$$

  Thus, if $\underline{x}'_i > \underline{x}_i$, the domain of x_i can be shrinked to $[\underline{x}'_i, \overline{x}_i]$ without loss of any global optima.
- similar theorems for $\underline{x}_i - x_i \leq 0$ and $g_i(x) \leq 0$.

OBR Issues

- Main issue: the available dual solution λ^* is an *approximation* ...

  if used in OBR ...

  ... OBR might *loose* the global optima !

- Solutions: two ways to take advantage of OBR
  1. build a proved dual solution (Kearfott) ...
  2. validate the reduction proposed by OBR with CP !

A proved dual solution (Kearfott)

- Prove existence of a solution of a system of equations combining
  - the dual of linear relaxation R (2)

$$\begin{array}{ll} max & b^T y \\ s.t. & A^T \lambda = d \end{array} \tag{4}$$

  - with the Kuhn-Tucker conditions which provides lower and upper bounds on R (2) and its dual (4):

$$(KT) \left\{ \begin{array}{ll} A^T \lambda - d & = 0 \\ \lambda_i (A_{i,:} x - b_i) & = 0, 1 \leq i \leq m \end{array} \right. \tag{5}$$

where $A_{i,:}$ is i-th row of A.

Kearfott's approach drawbacks

- Critical issue:
  - The system is overconstrained
    (→ cannot directly use existence theorems).
- To get a squared system
  → relax the relaxation (remove constraints)
  acceptable as all we need is a lower bound of λ^*
- Drawbacks:
  - Less efficient to prove the existence of a solution.
  - Bounds may be wide due to weakened relaxation.

CP approach: intuition

- Prove that no global solution is lost !
  - *Essential observation*: if the constraint system

$$\begin{aligned} & L \leq f(x) \leq U \\ & g_i(x) = 0,\ i = 1..k \\ & g_j(x) \leq 0,\ j = k+1..m \end{aligned} \tag{6}$$

    has *no solution* with x set to $[\underline{x}_i, \underline{x}'_i]$, then the reduction computed by OBR is valid
  - Task achieved by a *classical filtering process*
- Otherwise add this box to the list of boxes to process

CP approach: algorithm

```
$\mathcal{L}_r \leftarrow \emptyset$ % $\mathcal{L}_r$: set of potential non-solution boxes
for each variable $x_i$ do
    Apply OBR
        and add the generated potential non-solution boxes to $\mathcal{L}_r$
for each box $\mathrm{B_i}$ in $\mathcal{L}_r$ do
    $\mathrm{B_i'} \leftarrow 2B$-filtering$(\mathrm{B_i})$
    if $\mathrm{B_i'} = \emptyset$ then reduce the domain of $x_i$
    else $\mathrm{B_i''} \leftarrow$ Quad-filtering$(\mathrm{B_i'})$
            if $\mathrm{B_i''} = \emptyset$ then reduce the domain of $x_i$
            else add $\mathrm{B_i}$ to global list of box to be handled endif
    endif
```

Experimental Results (1)

- Compares 4 versions of the branch and bound algorithm:
  - without OBR
  - with unsafe OBR
  - with safe OBR based on Kearfott’s approach
  - with safe OBR based on CP techniques

  implemented with Icos using Coin/CLP and Coin/Ipopt.
- On 78 benches (from Ryoo & Sahinidis 1995, Audet thesis and the coconut library)
- All experiments have been done on PC-Notebook/1Ghz.

Experimental Results (2): Synthesis

Synthesis of the results :

	$\Sigma_t(s)$	$\%saving$
no OBR	2384.36	-
unsafe OBR	881.51	63.03%
safe OBR Kearfott	1975.95	17.13%
safe OBR CP	454.73	80.93%

(with a timeout of 500s)

Experimental Results (3): best cases

name	Safe OBR CP	Safe OBR Kearfott	% saving
himmel11	2.4	-	-
ex5_2_2_case3	4.49	-	-
c-chem7	10.5	-	-
ex2_1_5	3.52	15.64	77.49%
ex7_2_5	2.19	7.14	69.32%
ex14_2_5	2.48	7.19	65.50%
ex7_2_10	0.16	0.41	60.97%
ex8_1_6	0.73	1.54	52.59%
ex14_2_2	3.05	6.38	52.19%
ex7_3_1	3.38	5.95	43.19%

Experimental Results (4): worst cases

name	Safe ORB CP	Safe OBR Kearfott	% saving
ex2_1_1	0.25	0.21	-19.04%
ex9_1_9	0.11	0.09	-22.22%
ex2_1_4	1.25	1.01	-23.76%
ex9_2_8	0.05	0.04	-25%
c-chem18	4.83	3.57	-35.29%
ex3_1_2	0.19	0.14	-35.71%
c-audet147	0.61	0.44	-38.63%
c-audet140b	0.13	0.09	-44.44%
alkyl	32.45	20.55	-57.90%
ex5_2_2_case1	7.75	2.88	-169.09%

Conclusion

Constraint programming techniques

- allow a *safe* and *efficient* implementation of OBR
- can outperforms standard mathematical methods
- allow a safe embedding of OBR in a *simple* way thanks to refutation
- might be suitable for other unsafe method

The End !