Nonlinear Bi-Objective Optimization: Improving the Upper Envelope using Feasible Line Segments

Damir Aliquintui[1,b)], Ignacio Araya[1,a)], Franco Ardiles[1,c)] and Braulio Lobo[1,d)]

[1] *Pontificia Universidad Católica de Valparaíso, Escuela de Ingeniería Informática, Chile.*

[a)] Corresponding author: ignacio.araya@pucv.cl
[b)] damir.aliquintui.p@mail.pucv.cl
[c)] franco.ardiles.a@mail.pucv.cl
[d)] braulio.lobo.m@mail.pucv.cl

Abstract. In this work we propose a *segment-based representation* for the upper bound of the non-dominated set in interval branch & bound solvers for bi-objective non linear optimization. We also warranty that every point over the *upper segments* is dominated by at least one point in the feasible objective region. Segments are generated by linear envelopes of the image of *feasible* line-segments. The segment-based representation allows us to converge more quickly to the desired precision of the whole strategy.

INTRODUCTION

In this work we deal with Non-Linear Bi-Objective Optimization (NLBOO) problems defined by:

$$\begin{array}{ll} \min\limits_{x \in \boldsymbol{x}} & f_1(x), f_2(x) \\ \text{s.t} & g(x) \leq 0, h(x) = 0 \end{array} \tag{1}$$

with $x \in \mathbb{R}^n$ the set of variables varying in the box $\boldsymbol{x}^1$, $f = (f_1, f_2) : \mathbb{R}^n \rightarrow \mathbb{R}^2$ two real-valued objective functions, $g : \mathbb{R}^n \rightarrow \mathbb{R}^p$ the inequality constraints and $h : \mathbb{R}^n \rightarrow \mathbb{R}^q$ the equality constraints. Functions f, g, h can be non-linear.

A point x is said to be **feasible** or to be a *feasible solution* if it satisfies all the constraints. The **feasible region** $\mathcal{X} \subset \mathbb{R}^n$ consists of the set of all the feasible solutions. Its image $\mathcal{Y} := \{y : y_1 = f_1(x), y_2 = f_2(x), x \in \mathcal{X}\}$, is called the **feasible objective region**.

Let y and y' be two points in $\mathbb{R}^2$, we say that y **dominates** y' iff $y_1 \leq y'_1$, $y_2 \leq y'_2$ and $y \neq y'$. A feasible point $y \in \mathcal{Y}$ is a **non-dominated** point of $\mathcal{Y}$ if there is no other $y' \in \mathcal{Y}$ such that y' dominates y. The set of non-dominated points is denoted $\mathcal{Y}^*$. A feasible solution $x \in \mathcal{X}$ is **Pareto-optimal** if $y = f(x)$ is non-dominated. The set of Pareto-optimal solutions is denoted $\mathcal{X}^*$. A set $\mathcal{Y}$ of points in $\mathbb{R}^2$ is **dominance-free** if there is no $(y, y') \in \mathcal{Y}^2$, such that y dominates y'.

A posteriori methods attempt to find a *good representation* for the set of *Pareto-optimal solutions*. Finding a description, analytically or numerically, for the whole set of non-dominated points is a difficult task since this set includes typically a very large or infinite number of points. Several methods dealing with this problem are based on interval arithmetic and/or branch & bound strategies.

Interval branch & bound solvers (e.g., [1, 2]) typically starts with an initial box with the initial variable domains and build a search tree by *bisecting* and *filtering* variable domains. Bisection consists in splitting a box into two sub-boxes by dividing the domain of one variable and generating two child nodes. Filtering (or contraction) consists in removing inconsistent values from the bounds of the box. Solvers also include

[1] An interval $\boldsymbol{x}_i = [\underline{x_i}, \overline{x_i}]$ defines the set of reals x_i, such that $\underline{x_i} \leq x_i \leq \overline{x_i}$. $\text{mid}(\boldsymbol{x}) = \frac{\underline{x_i}+\overline{x_i}}{2}$ denotes the midpoint of the interval $\boldsymbol{x}$. A box $\boldsymbol{x}$ is a Cartesian product of intervals $\boldsymbol{x}_1 \times ... \times \boldsymbol{x}_i \times ... \times \boldsymbol{x}_n$.

upper-bounding procedures for finding good feasible solutions in each processed box. As output they return a dominance-free set of solutions which approximates the set of Pareto-optimal solutions and a thin envelope, in the objective region, which *certainly* contains the set of non-dominated points $\mathcal{Y}^*$.

Algorithm 1 shows the structure of a standard interval branch & bound solver (based on [2]). Before calling the procedure, we must include the additional constraints $y_1 = f_1(x)$ and $y_2 = f_2(x)$ into h, where $y \in \mathbb{R}^2$ is the vector of the objective variables. The algorithm takes as input the NLBOO problem P, a composed box $(\boldsymbol{x}^0, \boldsymbol{y}^0)$ with the initial domains of the decision and objective variables and an user-defined precision ϵ.

procedure *NLBOO-Solver* $(P = (f, g, h), \boldsymbol{x}^0, \boldsymbol{y}^0, \epsilon)$; **out:** $\mathcal{X}', \mathcal{Y}'$
    $\mathcal{S} \leftarrow \{(\boldsymbol{x}^0, \boldsymbol{y}^0)\}$; $\mathcal{Y}' \leftarrow \{\}$;
    while $\mathcal{S} \neq \emptyset$ **do**
        $(\boldsymbol{x}, \boldsymbol{y}) \leftarrow$ select and remove a node from $\mathcal{S}$;
        $(\boldsymbol{x}, \boldsymbol{y}) \leftarrow$ filtering$((\boldsymbol{x}, \boldsymbol{y}), \mathcal{Y}', P, \epsilon)$;
        if $\boldsymbol{x} \neq \emptyset$ **then**
            $(\mathcal{X}', \mathcal{Y}') \leftarrow$ upper-bounding$(\boldsymbol{x}, \boldsymbol{y}, \mathcal{X}', \mathcal{Y}', P)$;
            $(\boldsymbol{x}^l, \boldsymbol{y}^l, \boldsymbol{x}^r, \boldsymbol{y}^r) \leftarrow$ bisect $(\boldsymbol{x}, \boldsymbol{y}, P)$;
            $\mathcal{S} \leftarrow \mathcal{S} \cup \{(\boldsymbol{x}^l, \boldsymbol{y}^l), (\boldsymbol{x}^r, \boldsymbol{y}^r)\}$;

Algorithm 1: Algorithm NLBOO-solver

The search starts with the initial box and builds a search tree represented by the set of leave nodes $\mathcal{S}$. In each iteration, a node is selected from $\mathcal{S}$ and treated by filtering and upper-bounding procedures. The upper-bounding procedure attempts to find good feasible solutions for improving the *upper envelope* of the non-dominated points. The upper envelope is represented by a set of dominance-free points $\mathcal{Y}'$. Thus, feasible points found by the upper-bounding procedure update $\mathcal{Y}'$ maintaining the dominance-free property. The set $\mathcal{X}'$ in turn keeps the feasible solutions associated to the points in $\mathcal{Y}'$.

The filtering method attempts to remove inconsistent values from the bounds of the box. Some filtering methods take into account the set $\mathcal{Y}'$ for filtering ϵ-dominated solutions (e.g., discarding tests [1, 3], dominance contractors [2]). If the box has not been discarded by the filtering process, then it is bisected and its children are added into $\mathcal{S}$. The procedure iterates until all the boxes are discarded, meaning that the image of any feasible solution of the problem is ϵ-dominated by at least one point in $\mathcal{Y}'$.

As output, the solver returns a set of feasible solutions $\mathcal{X}' \subset \mathbb{R}^n$ with their images $\mathcal{Y}' \subset \mathbb{R}^2$ *providing* that any non-dominated point in $\mathcal{Y}^*$ is ϵ-dominated[2] by some point in $\mathcal{Y}'$.

THE SET $\mathcal{Y}'$ AND THE UPPER-BOUNDING PROCEDURE

In any moment of the search, the set $\mathcal{Y}'$ defines the current ϵ-dominated region. Thus, together with the constraint $y = f(x)$, $\mathcal{Y}'$ can be used for reducing the search space by discarding ϵ-dominated solutions. For instance see Fig. 1-left. Points represent the set $\mathcal{Y}'$ in the objective space. The region in light grey is ϵ-dominated by these points. The lower bound $(\underline{y_1}, \underline{y_2})$ of the box A, is dominated by the second point (from left to right) in $\mathcal{Y}'$, thus the box A can be discarded. The *dominance peeler* method, proposed in [2], allows us to contract a box $\boldsymbol{y}$ when the points $(\underline{y_1}, \overline{y_2})$ or $(\overline{y_1}, \underline{y_2})$ are ϵ-dominated by points in $\mathcal{Y}'$. For example, the box B in the figure can be reduced to the solid line box.

Achieving a good set $\mathcal{Y}'$ in early stages of the search is important for reducing the size of the search tree. For that reason, upper-bounding is a crucial component of the solver. It should offer good solutions in a short period of time. Most of interval-based approaches (e.g., [1, 2, 3]) propose simply to use the midpoint of $\boldsymbol{x}$, i.e, to test the feasibility of mid(x) and add $y = f(\text{mid}(x))$ into $\mathcal{Y}'$ if y is not dominated by any other point in the set. The method is fast and effective: as the search progress, the set $\mathcal{Y}'$ gradually improves until reaching the desired precision. The method alone, however, does not offer a good approximation of the non-dominated set in the current box. In fact, we are required to process a reasonable amount of boxes in order to obtain an *initial* approximation of the whole set of non-dominated points $\mathcal{Y}^*$.

[2]Let y and y' be two points in $\mathbb{R}^2$, we say that y **ϵ-dominates** y' iff $y_1 \leq y'_1 + \epsilon$, $y_2 \leq y'_2 + \epsilon$.

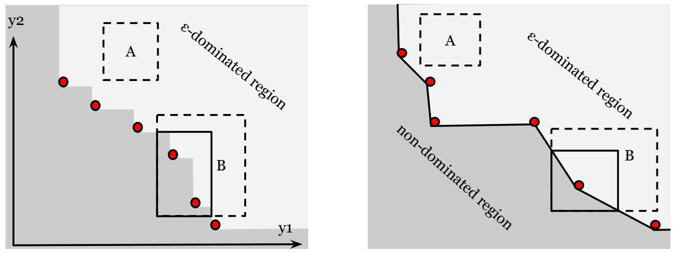

FIGURE 1. Representations of the upper envelope: (left) by a set of dominance-free feasible points; (right) by a set of line segments.

We propose to represent the upper envelope of non-dominated points by using a set of line segments instead of feasible points. These segments can be horizontal, vertical or even *oblique* lines as shown in Fig. 1-right. Segments are represented by its end points (points in the figure). Remark that these points do not necessarily correspond to feasible solutions. They simply delimit the region where the non-dominated points are. We also propose an upper-bounding algorithm for finding upper line segments such that, every point in a segment corresponds to a non-dominated point or it is dominated by some feasible point in the objective region.

A CHEAP ALGORITHM FOR FINDING UPPER LINE SEGMENTS

The objective of our upper-bounding algorithm is first, obtaining a feasible curve for approximating the set of non-dominated points in the box and then, generating an upper line segment passing just above this curve. To do this, the algorithm first constructs a convex polytope inside the current box $\boldsymbol{x}$, such that all the points inside the polytope satisfy the original constraint system (for more details about the construction of an *inner polytope* refer to [4]). Then, by using a simplex algorithm, we find two feasible solutions ($x^{(1)}$ and $x^{(2)}$) in the polytope, each of them minimizing a linearization of one of the objective functions [3]. The images $y^{(1)}$ and $y^{(2)}$ of these points, give us an approximation of the most left and the lowest feasible points in the objective space respectively (see Fig. 2).

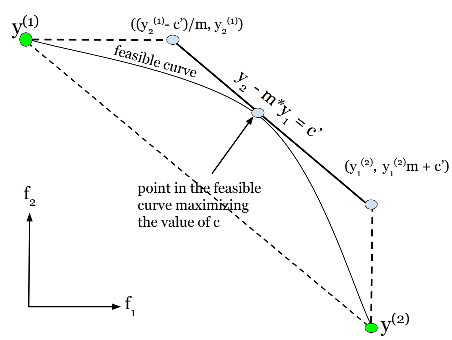

FIGURE 2. The upper bounding procedure returns two feasible points $y^{(1)}$ and $y^{(2)}$ and an upper line segment.

As both solutions are in the same convex polytope, we know that all the points in the line connecting $x^{(1)}$ and $x^{(2)}$ are also feasible solutions. Thus, their images will offer an approximation of the set of non-dominated points in the box (feasible curve in Fig. 2). Note that due to the objective functions may be non-linear, the image of a line in $\mathcal{X}$ is not necessarily a line in the objective space.

The next step consists in finding a line segment passing over any point in the feasible curve. In other words, we want to find a line $y_2 - my_1 = c'$ such that $c' \geq f_2(x) - m \cdot f_1(x)$ for all x in the line connecting $x^{(1)}$

[3] A linearization of an objective function f can be obtained by a first-order Taylor series approximation about the midpoint x' of the box, i.e., $f(x) \approx f(x') + \nabla f(x')^T \cdot (x - x')$

and $x^{(2)}$ (see the figure). For finding a minimal c' we solve the uni-variate nonlinear program $\max_{t\in[0,1]} f_1(x(t)) - mf_2(x(t))$, where $x(t) = x^{(1)} + (x^{(2)} - x^{(1)})t$ and $m = (y_2^{(2)} - y_2^{(1)})/(y_1^{(2)} - y_1^{(1)})$. Note that we use the same slope m of the line connecting the points $y^{(1)}$ and $y^{(2)}$. The problem is solved by interleaving bisections of the interval $\boldsymbol{t}$ and uni-variate Interval Newton steps which filter sub-optimal solutions and find new feasible upper bounds. The method returns an overestimation c' of the optimal value which is used for generating the upper line segment: $(\frac{y_2^{(1)}-c'}{m}, y_2^{(1)}); (y_1^{(2)}, my_1^{(2)} + c')$. Finally, the feasible points $y^{(1)}$ and $y^{(2)}$ and the generated upper line segment update the upper envelope $\mathcal{Y}'$ which is represented by a set of line segments.

FIRST RESULTS

Table 1 shows a comparison between a standard interval-based solver which represents the upper envelope by using a dominance-free set of points (STD) and our approach which represents the upper envelope by using line segments (LINESEG). Both algorithms were implemented in the Ibex library [5]. They use state-of-the-art optimization techniques such as the dominance peeler method[4] and other contractors commonly used by interval-based solvers for single objective optimization [6]. As LINESEG, the upper-bounding in STD also uses the points $y^{(1)}$ and $y^{(2)}$ for updating $\mathcal{Y}'$ but without generating the upper line segment. For each instance and strategy, we report the given precision ϵ, the spent CPU time, the number of processed boxes and the size of the set $\mathcal{Y}'$. We run the strategies in the set of 13 instances described in [2]. The reported results are from those instances who have the largest differences in time.

TABLE 1. Comparison between STD and LINESEG.

		LINESEG			STD		
Instance	ϵ	time	#box	$\lvert\mathcal{Y}'\rvert$	time	#box	$\lvert\mathcal{Y}'\rvert$
binh	0.1	**0.22**	157	642	1.76	1205	2801
osy	0.1	**5.35**	1757	2225	5.82	2082	2513
kim	0.1	3.01	1583	710	**2.72**	2494	1501
mop-7	0.5	**11.4**	4395	547	18.2	5498	473
nbi	0.1	**1.16**	340	175	1.68	503	319
tan	0.001	**0.41**	442	591	0.98	888	1035

Note that the results are promising. The approach outperforms the standard strategy in most of the instances. As a future work, we plan to go further and generate more upper line segments for approximating better the feasible curve in Fig.2.

REFERENCES

[1] J. Fernández and B. Tóth, "Obtaining the efficient set of nonlinear biobjective optimization problems via interval branch-and-bound methods," *Computational Optimization and Applications*, vol. 42, no. 3, pp. 393–419, 2009.

[2] B. Martin, A. Goldsztejn, L. Granvilliers, and C. Jermann, "Constraint propagation using dominance in interval branch & bound for nonlinear biobjective optimization," *European Journal of Operational Research*, vol. 260, no. 3, pp. 934–948, 2017.

[3] A. Goldsztejn, F. Domes, and B. Chevalier, "First order rejection tests for multiple-objective optimization," *Journal of Global Optimization*, vol. 58, no. 4, pp. 653–672, 2014.

[4] I. Araya, G. Trombettoni, B. Neveu, and G. Chabert, "Upper bounding in inner regions for global optimization under inequality constraints," *Journal of Global Optimization*, pp. 1–20, 2014.

[5] G. Chabert and L. Jaulin, "Contractor Programming," *Artificial Intelligence*, vol. 173, pp. 1079–1100, 2009.

[6] I. Araya, G. Trombettoni, and B. Neveu, "A contractor based on convex interval taylor," in *Integration of AI and OR Techniques in Contraint Programming for Combinatorial Optimzation Problems*, pp. 1–16, Springer, 2012.

[4]See, as an example of this method, the reduction of the box B in Fig. 1 (left and right).