Using a B&B Algorithm from Multiobjective Optimization to Solve Constrained Optimization Problems

Gabriele Eichfelder[1,b)], Kathrin Klamroth[2,c)] and Julia Niebling[1,a)]

[1] *Institute for Mathematics, Technische Universität Ilmenau, Po 10 05 65, D-98684 Ilmenau, Germany.*
[2] *School of Mathematics and Natural Sciences, University of Wuppertal, Gaußstr. 20, D-42119 Wuppertal, Germany.*

[a)] Corresponding author: julia.niebling@tu-ilmenau.de
[b)] gabriele.eichfelder@tu-ilmenau.de
[c)] klamroth@math.uni-wuppertal.de

Abstract. Multiobjective optimization problems are often solved using methods from scalar-valued optimization: One – or a series of – associated scalarizations are formulated and solved using appropriate methods from scalar-valued optimization. Conversely, tools from multiobjective optimization can be applied in combination with appropriate filtering techniques to approach scalar-valued optimization problems. This shows that both topics have a substantial relationship which can be exploited for mutual benefit. In this work we analyze this relationship and suggest to adapt a deterministic multiobjective branch-and-bound algorithm (originally designed for nonconvex multiobjective optimization problems) to solve scalar-valued optimization problems with nonconvex constraints.

1 MOTIVATION

Solving constrained optimization problems is a prevalent task in practical and theoretical problems. In practical applications of constrained optimization, it is often observed that (some of) the constraints are soft, i.e., slight violations may be acceptable, and improvements beyond the given bound may have extra benefits. If optimization methods are applied in a straight-forward way to such problems, soft constraints are generally not well-represented. In this respect, multiobjective optimization methods have a clear advantage since they allow to explicitly model the trade-off between constraint satisfaction and objective optimization. This motivates to further analyze the relationship between multiobjective optimization and constrained optimization problems.

But even in instances of constrained optimization problems that do not have any soft constraints, a multiobjective perspective may be beneficial. To see this, we consider the following optimization problem, which was studied in [1].

Example 1.1 *We study the scalar-valued optimization problem*

$$\begin{array}{rl} \min & f(x) = x_1 - x_2 \\ \text{s.t.} & g(x) = -x_1^2 - (x_2 - 5)^2 + 25 + \sqrt{2} \leq 0, \\ & x \in [1, 2] \times [0, 1]. \end{array} \tag{1}$$

This optimization problem has the globally optimal solution $x^* = (\sqrt[4]{2}; 0)^T \approx (1.1892; 0)^T$ *with the minimal value* $f(x^*) = \sqrt[4]{2} \approx 1.1892.$

In the work of Kirst, Stein and Steuermann, [1], it is briefly explained why the basic αBB-method, which was introduced in [2], does not find any feasible point of (1). Even though this optimization problem is rather simple, the basic αBB-method fails since no upper bounds on the minimal value are found. The difficulty of finding feasible points is a rather frequent problem, even in optimization problems modeling practical problems.

The αBB-method, [2], is a well-known deterministic global optimization method for scalar-valued optimization problems. The main steps are to construct upper and lower bounds for the globally minimal value and to improve these in a branch-and-bound framework. Typically, upper bounds are obtained as images of some feasible points which are found by the algorithm mostly automatically. Lower bounds are obtained by minimizing a convex underestimator of the objective function over a convex superset of the feasible set within a considered subbox. Note that a convex underestimator is a convex function which is pointwise smaller than the original function. A solution of the relaxed and convexified optimization problem can serve as an upper bound by applying the original objective function to it but only if it is feasible for the original problem. By updating the upper and lower bounds during the algorithm we can narrow down on the globally minimal value until a given accuracy is reached.

In the case of problem (1) above, the αBB method does not progress because solving the convexified problem always leads to infeasible points, see [1] for a more detailed explanation. Indeed, some lower bounds can be constructed, but an upper bound is never obtained. Therefore, the algorithm does not find an approximation of the globally minimal value with a satisfactory accuracy and thus it will never terminate.

In this work we sketch some relations between constrained optimization and multiobjective optimization in order to derive new approaches for the solution of scalar-valued constrained optimization problems. The basic idea is not new. For example, in the context of genetic algorithms constraints are often included into the objective function(s) in order to penalize infeasible solutions, see, e.g., [3]. Conversely, solving multiobjective problems by using scalarization approaches is a well established method and shows the deep correspondence between scalar-valued and multiobjective optimization. From a more theoretical perspective, for example Klamroth and Tind [4] examined the relations between constrained and multiobjective optimization in a very detailed way. They consider many different scalarization approaches like the e-constraint method or the weighted-sum approach and set them in relation to relaxations and penalization strategies known from constrained optimization.

2 DEFINITIONS OF GLOBAL AND MULTIOBJECTIVE OPTIMIZATION

Since we are working with multiobjective optimization problems and have to compare vectors we define the following order relations for two vectors $x, y \in \mathbb{R}^m$: $x \leqq y$ if and only if $x_j \leq y_j$ for all $j = 1, \ldots, m$; $x \leq y$ if and only of $x_j \leq y_j$ for all $j = 1, \ldots, m$ and $x \neq y$; and $x < y$ if and only if $x_j < y_j$ for all $j = 1, \ldots, m$. The symbols $\geqq, \geq$ and $>$ are defined analogously.

Throughout this paper we want to consider two optimization problems. Let $X \subseteq \mathbb{R}^n$ be a *box*, i.e. there are vectors $\underline{x}, \overline{x} \in \mathbb{R}^n$ with $\underline{x} \leq \overline{x}$ and $X := [\underline{x}, \overline{x}] := \{x \in \mathbb{R}^n \mid \underline{x} \leqq x \leqq \overline{x}\}$. We denote the set of all n-dimensional real-valued boxes by $\mathbb{IR}^n$. Furthermore, let $f\colon \mathbb{R}^n \to \mathbb{R}$ and $g_k\colon \mathbb{R}^n \to \mathbb{R}$, $k = 1, \ldots, p$ be twice continuously differentiable functions. The first problem (P) is a constrained optimization problem and the second one (MOP) is the associated multiobjective problem:

$$\begin{aligned} &\min_{x \in X} f(x) \\ &\text{s.t } g_k(x) \leq 0,\ k = 1, \ldots, p \end{aligned} \qquad (P) \qquad\qquad \min_{x \in X} \begin{pmatrix} f(x) \\ g_1(x) \\ \vdots \\ g_p(x) \end{pmatrix} \qquad (MOP)$$

We assume that the feasible set S of (P) given by $S := \{x \in X \mid g_k(x) \leq 0, k = 1, \ldots, p\}$ is non-empty. The following definitions are given for (P):

Definition 2.1 *Let $\varepsilon > 0$ be given.*

- *A solution $x^* \in S$ is said to be minimal for (P) if $f(x^*) \leq f(x)$ for all $x \in S$.*
- *A solution $x^* \in S$ is said to be ε-minimal for (P) if $f(x^*) < f(x) + \varepsilon$ for all $x \in S$.*

In multiobjective optimization several objective functions have to minimized simultaneously. In general there does not exist a solution that minimizes all objectives at the same time. We use the classical concept of *efficiency* (see, for example, [5] for an introduction to multiobjective optimization). We denote the objective function by $h\colon \mathbb{R}^n \to \mathbb{R}^{1+p}$ with $h(x) = (f(x), g_1(x), \ldots, g_p(x))^T$. Moreover $e = (1, \ldots, 1)^T \in \mathbb{R}^{p+1}$ is the $(p+1)$-dimensional all one vector.

Definition 2.2 *Let $\varepsilon > 0$ be given.*

- *A solution $x^* \in X$ is said to be efficient for (MOP) if there does not exist another $x \in X$ with $h(x) \leq h(x^*)$.*
- *A solution $x^* \in X$ is said to be weakly efficient for (MOP) if there does not exist another $x \in X$ with $h(x) < h(x^*)$.*
- *A solution $x^* \in X$ is said to be ε-efficient for (MOP) if there does not exist another $x \in X$ with $h(x) \leq h(x^*) - \varepsilon e$.*

3 RELATIONSHIP BETWEEN (P) AND (MOP)

Note that for the optimization problems (P) and (MOP) an optimal/efficient solution exists, respectively, because both feasible sets S and X are compact sets and the objective functions are continuous. Moreover, for the efficient set of (MOP) external stability holds since the ordering cone $\mathbb{R}^{p+1}_+$ is a pointed closed convex cone and $h(X)$ is a compact set, compare with [6, Theorem 3.2.9].

We begin with some obvious relations between both optimization problems.

Lemma 3.1 *Let (P) and (MOP) be given. Then it holds:*

- $x \in X$ *is feasible for (P)* $\Rightarrow$ $x \in X$ *is feasible for (MOP)*
- $x \in X$ *is efficient for (MOP) with* $g_k(x) \leq 0$ *for all* $k = 1, \dots, p \Rightarrow x \in X$ *is feasible for (P)*

Moreover we can state a relationship between ε-optimal solutions of (P) and ε-efficient solutions of (MOP).

Lemma 3.2 *Let* $x^* \in S$ *be* ε*-optimal for (P). Then* x^* *is* ε*-efficient for (MOP).*

The next theorem, which is given for instance in [4], also holds for our setting.

Theorem 3.3 *The set of optimal solutions of the constrained problem (P) always contains an efficient solution of the associated multiobjective problem (MOP), and all optimal solutions of (P) are weakly efficient for (MOP). Conversely, the set of efficient solutions of* (MOP) *contains at least one optimal solution of* (P).

As explained in [4], an optimal solution of (P) can be calculated as a specific efficient solution of (MOP). In case that the complete efficient set of (MOP) is known we can choose those efficient solutions $x^* \in S$ with the smallest value $f(x^*)$. These solutions would then also be optimal for (P). But in general the determination of the complete efficient set is impractical. However, if an algorithm can compute an approximation of the efficient set then it is possible to find near-optimal solutions of the constrained optimization problem.

4 A BRANCH-AND-BOUND ALGORITHM FOR MULTIOBJECTIVE OPTIMIZATION PROBLEMS

To solve (MOP) we use the novel branch-and-bound based algorithm for nonconvex multiobjective optimization which was given in [7]. This algorithm is then adapted for our special purpose to solve the original constrained scalar-valued optimization problem. We commence by summarizing the main idea of the algorithm in [7] for a vector-valued objective function $h\colon \mathbb{R}^n \to \mathbb{R}^m$ in Algorithm 1 below, see also [8]:

Algorithm 1

INPUT: $X \in \mathbb{R}^n, h \in C^2(\mathbb{R}^n, \mathbb{R}^m)$
OUTPUT: $\mathcal{L}_S$

$\mathcal{L}_W \leftarrow \{X\}, \mathcal{L}_S \leftarrow \emptyset$
while $\mathcal{L}_W \neq \emptyset$ **do**
  Select a box X^* from $\mathcal{L}_W$ and delete it from $\mathcal{L}_W$ *Selection rule*
  Bisect X^* perpendicularly to a direction of maximum width $\to X^1, X^2$
  for $l = 1, 2$ **do**
    if X^l cannot be discarded **then** *Discarding tests*
      if X^l satisfies a termination rule **then** *Termination rule*
        Store X^l in $\mathcal{L}_S$
      else Store X^l in $\mathcal{L}_W$

The lists $\mathcal{L}_W$ and $\mathcal{L}_S$ are the working list and the solution list, respectively.

Using the novel ideas on discarding tests and a new termination procedure as suggested in [7], this branch-and-bound algorithm yields a set of ε-efficient points. In order to adapt the algorithm from [7] for the solution of the constrained optimization problem (P), a filtering step that searches for a solution x^* with the smallest first objective value $h_1(x^*) = f(x^*)$ and with $h_k(x^*) \leq 0$ for all $k = 2, \dots, p+1$ is applied as a postprocessing. We implemented the new algorithm as well as the original algorithm of [7] in Matlab.

Example 4.1 *We consider again the optimization problem from Example 1.1 and its associated multiobjective optimization problem*

$$\min \begin{pmatrix} f(x) \\ g(x) \end{pmatrix} = \begin{pmatrix} x_1 - x_2 \\ -x_1^2 - (x_2 - 5)^2 + 25 + \sqrt{2} \end{pmatrix} \ s.t.\ x \in [1, 2] \times [0, 1]. \tag{2}$$

After the algorithm from [7] terminates, the filtering step described above is applied. We tested two different values of $\varepsilon > 0$. Both computed ε-efficient solutions and their objective values are given in Table 1. Indeed, the computed solutions from (2) are ε-minimal solutions of the original scalar-valued problem (1).

TABLE 1. Computed ε-efficient solutions and minimal values of (2) by using the algorithm from [7] and a filtering step

ε	**ε-efficient solution**	**minimal value**	**time [s]**
0.1	$(1.25; 0.0139)^T$	$(1.2361; -0.0092)^T$	19.9251
0.01	$(1.1914; 0)^T$	$(1.1914; -0.0052)^T$	194.3867

Note that the basic algorithm generally computes too many ε-efficient solutions of the associated multiobjective optimization problem, most of which are not of interest because at least one of their objective values $h_2, \ldots, h_{p+1}$ is considerably larger than 0. To overcome this difficulty, the approach should be adapted for the solution of constrained optimization problems. New selection and termination rules are currently developed and tested, and will be the topic of the forthcoming talk. Other discarding tests are useful as well to discard irrelevant boxes as early as possible.

5 CONCLUSIONS AND OUTLOOK

In this work we presented and used some basic relations between constrained scalar-valued optimization problems and multiobjective optimization problems. These relations were used to overcome known difficulties in the solution of nonconvex constrained optimization problems by using a deterministic branch-and-bound framework for multiobjective optimization. The goal is to use and adapt this branch-and-bound algorithm for nonconvex multiobjective optimization. Applying the basic algorithm with some filtering steps afterwards already gives very promising results. Improvements involving better selection and termination rules can reduce the numerical effort significantly.

6 ACKNOWLEDGMENTS

The third author thanks Carl-Zeiss-Stiftung and DFG-funded Research Training Group 1567 ”Lorentz Force Velocimetry and Lorentz Force Eddy Current Testing” for financial support.

REFERENCES

[1] P. Kirst, O. Stein, and P. Steuermann, “Deterministic upper bounds for spatial branch-and-bound methods in global minimization with nonconvex constraints.” in *TOP*, 23(2) (2015), pp. 591–616.

[2] C. Adjiman, S. Dallwig, C. Floudas, and A. Neumaier, “A Global Optimization Method, αBB, for General Twice-Differentiable Constrained NLPs: I - Theoretical Advances.” in *Computers & Chemical Engineering*, 22(9) (1998), pp. 1137–1158.

[3] E. Camponogara and S. Talukdar, “A Genetic Algorithm for Constrained and Multiobjective Optimization.” in *3rd Nordic Workshop on Genetic Algorithms and Their Applications (3NWGA)* (1997), pp. 49–62.

[4] K. Klamroth and J. Tind, “Constrained optimization using multiple objective programming.” in *Journal of Global Optimization*, 37(3) (2007), pp. 325–355.

[5] M. Ehrgott, *Multicriteria optimisation*, 2nd ed. (Springer-Verlag Berlin Heidelberg, 2005).

[6] H. N. Y. Sawaragi and T. Tanino, *Theory of Multiobjective Optimization* (Academic Press, 1985).

[7] J. Niebling and G. Eichfelder, “A Branch-and-Bound based Algorithm for Nonconvex Multiobjective Optimization,” in *Preprint-Series of the Institute for Mathematics, Technische Universität Ilmenau, Germany*, 18-03 (2018).

[8] J. Fernández and B. Tóth, “Obtaining the efficient set of nonlinear biobjective optimization problems via interval branch-and-bound methods.” in *Computational Optimization and Applications*, 42(3) (2009), pp. 393–419.