

Article

The Pareto Tracer for General Inequality Constrained Multi-Objective Optimization Problems

Fernanda Beltrán ¹, Oliver Cuate ^{1,2}  and Oliver Schütze ^{1,*}

¹ Computer Science Department, Cinvestav-IPN, 07360 Mexico City, Mexico; mbeltran@computacion.cs.cinvestav.mx (F.B.); ocuate@computacion.cs.cinvestav.mx (O.C.)

² ESFM, Instituto Politécnico Nacional, 07738 Mexico City, Mexico

* Correspondence: schuetze@cs.cinvestav.mx

Received: 22 October 2020; Accepted: 18 December 2020; Published: 20 December 2020



Abstract: Problems where several incommensurable objectives have to be optimized concurrently arise in many engineering and financial applications. Continuation methods for the treatment of such multi-objective optimization methods (MOPs) are very efficient if all objectives are continuous since in that case one can expect that the solution set forms at least locally a manifold. Recently, the Pareto Tracer (PT) has been proposed, which is such a multi-objective continuation method. While the method works reliably for MOPs with box and equality constraints, no strategy has been proposed yet to adequately treat general inequalities, which we address in this work. We formulate the extension of the PT and present numerical results on some selected benchmark problems. The results indicate that the new method can indeed handle general MOPs, which greatly enhances its applicability.

Keywords: multi-objective optimization; Pareto Tracer; continuation; constraint handling

1. Introduction

In many real-world applications, the problem occurs that several conflicting and incommensurable objectives have to be optimized concurrently. As general example, in the design of basically any product, both cost (to be minimized) and quality (to be maximized) are relevant objectives, among others. Problems of that kind are termed multi-objective optimization problems (MOPs). In the case all of the objectives are continuous and in conflict with each other, it is known that there is not one single solution to be expected (as it is the case for scalar optimization problems, i.e., problems where one objective is considered) but an entire set of solutions. More precisely, one can expect that the solution set—the Pareto set, and, respectively its image, the Pareto front—forms at least locally an object of dimension $k - 1$, where k is the number of objectives involved in the problem. Due to this, “curse of dimensionality” problems with more than, e.g., four objectives are also called many objective optimization problems (MaOPs).

In the literature, many different methods for the numerical treatment of MOPs and MaOPs can be found (see also the discussion in the next section). One class of such methods is given by specialized continuation methods that take advantage of the fact that the solution set forms—at least locally and under certain mild assumption on the model as discussed in [1]—a manifold. Continuation methods start with one (approximate) solution of the problem and perform a movement along the Pareto set/front of the given M(a)OP via considering the underdetermined system of equations that is developed out of the Karush–Kuhn–Tucker (KKT) equations of the problem. By construction, continuation methods are of local nature. That is, if the Pareto set consists of different connected components, such methods will have to be fed with several starting points in order to obtain approximations of the entire solution set. On the other hand, continuation methods are probably most effective locally (i.e., within each connected component). Thus far, several multi-objective continuation methods have been proposed. Most of these continuation methods, however, are designed for or

restricted to the treatment of bi-objective problems (i.e., MOPs with two objectives). The method of Hillermeier [1] and the Pareto Tracer (PT [2]) have been proposed for general number k of objectives. The method of Hillermeier is applicable to unconstrained and equality constrained MOPs, and the PT in addition to box constrained problems. Thus far, no extensions for these two methods are known for the treatment of general inequalities, which represents a significant shortcoming since such constraints naturally arise in many applications (e.g., [3,4]). In this paper, we extend the PT for the treatment of general inequality constraints. To this end, we utilize and adapt elements from active set methods to decide which of the inequalities have to be treated as equalities at each candidate solution. We demonstrate the strength of the novel algorithm on several benchmark test functions and present comparisons to some other numerical multi-objective solvers. The results indicate that the new method can indeed reliably handle MOPs with general constraints.

The remainder of this paper is organized as follows. In Section 2, we shortly present the required background for the understanding of this work. In Section 3, we adapt the Pareto Tracer for the treatment of general (equality and inequality) constraints. In Section 4, we present some results of the PT as well as some other multi-objective numerical methods on selected benchmark problems. Finally, we draw our conclusions in Section 5 and give possible paths for future research.

2. Background and Related Work

In this section, we briefly state the main concepts and notations that are used for the understanding of this work (for details, we refer to, e.g., [5,6]).

We consider here continuous multi-objective optimization problem (MOPs) that can be defined mathematically as

$$\begin{aligned} \min_x \quad & F(x), \\ \text{s.t.} \quad & h_i(x) = 0, \quad i = 1, \dots, p, \\ & g_i(x) \leq 0, \quad i = 1, \dots, m, \end{aligned} \quad (1)$$

where $F : Q \subset \mathbb{R}^n \rightarrow \mathbb{R}^k$, $F(x) = (f_1(x), \dots, f_k(x))^T$ is the map of the k individual objectives $f_i : Q \subset \mathbb{R}^n \rightarrow \mathbb{R}$. We assume that all objectives and constraint functions are twice continuously differentiable. The domain Q of the functions is defined by the equality and inequality constraints of (1):

$$Q := \{x \in \mathbb{R}^n : h_i(x) = 0, i = 1, \dots, p \text{ and } g_i(x) \leq 0, i = 1, \dots, m\}. \quad (2)$$

If a point $x \in \mathbb{R}^n$ satisfies all constraints of (1), i.e., if $x \in Q$, we call this point feasible. Points $x \notin Q$ are called infeasible. If $k = 2$ objectives are considered, the problem is also termed a bi-objective optimization problem (BOP).

We say that a point $x \in Q$ dominates a point $y \in Q$ (in short: $x \prec y$) if $f_i(x) \leq f_i(y)$ for all $i = 1, \dots, k$, and there exists an index j such that $f_j(x) < f_j(y)$. A point x^* is called Pareto optimal or simply optimal if there does not exist a vector $y \in Q$ that dominates x^* . A point $x^* \in Q$ is called locally optimal if there does not exist a vector $y \in Q \cap N(x^*)$ that dominates x^* , where $N(x^*)$ is a neighborhood of x^* . The set P_Q of all Pareto optimal solutions is called the Pareto set, and its image $F(P_Q)$ the Pareto front. In [1], it has been shown that one can expect that both Pareto set and front typically form $(k - 1)$ -dimensional objects under certain (mild) conditions on the problem.

If all objectives and constraint functions are differentiable, local optimal solutions can be characterized by the Karush–Kuhn–Tucker (KKT) equations [7,8]:

Theorem 1. Suppose that x^* is locally optimal with respect to (1). Then, there exist Lagrange multipliers $\alpha \in \mathbb{R}^k$, $\lambda \in \mathbb{R}^p$ and $\gamma \in \mathbb{R}^m$ such that the following conditions are satisfied

$$\sum_{i=1}^k \alpha_i \nabla f_i(x^*) + \sum_{i=1}^p \lambda_i \nabla h_i(x^*) + \sum_{i=1}^m \gamma_i g_i(x^*) = 0 \quad (3a)$$

$$h_i(x^*) = 0, \quad i = 1 \dots p, \quad (3b)$$

$$g_i(x^*) \leq 0, \quad i = 1 \dots m, \quad (3c)$$

$$\alpha_i \geq 0, \quad i = 1 \dots k, \quad (3d)$$

$$\sum_{i=1}^k \alpha_i = 1, \quad (3e)$$

$$\gamma_i \geq 0, \quad i = 1 \dots m, \quad (3f)$$

$$\gamma_i g_i(x^*) = 0, \quad i = 1 \dots m. \quad (3g)$$

Multi-objective optimization is an active field of research, and thus far many numerical methods have been proposed for the treatment of such problems. There exist for instance many methods that are designed to compute single solutions such as the weighted sum method [9], the ϵ -constraint method [5,10], the weighted metric and weighted Tchebycheff method [5,11,12], as well as reference point problems [13–15]. All of these methods transform the given MOP into a scalar optimization problem (SOP) that can to a certain extent to include users' preferences. These methods can either be used as standalone algorithm (i.e., for the computation of single solutions) or be used to obtain a finite size approximation of the entire Pareto set/front of the given MOP via utilizing a clever sequence of these SOPs [5,16–19].

Further, there exist set oriented methods such as cell mapping techniques [20–23]), subdivision techniques [24–27], and multi-objective evolutionary algorithms (MOEAs, e.g., [3,28–34]). All of these methods manipulate an entire set of candidate solutions in each iteration and hence yield a finite size approximation of the solution set in one run of the algorithm. Hybridizations of such techniques with mathematical programming techniques can be found in [31,35–41].

Finally, a third class of numerical solvers for MOPs is given by specialized continuation methods that take advantage of the fact that the Pareto set/front of a given problem forms at least locally a manifold of a certain dimension. Methods of this kind start with a given (approximate) solution and perform a movement along the Pareto set/front of the problem. The first such method is proposed in [1], which can be applied to unconstrained and equality constrained MOPs of any number k of objectives, while no strategies are reported on how to treat inequalities. ParCont [42,43] is a rigorous predictor–corrector method that is based on interval analysis and parallelotope domains. The method can deal with equality and inequality constraints, but it is restricted to bi-objective problems. This restriction also holds for the method presented in [44], which has been designed to provide an equispaced approximation of the Pareto front. The Zigzag method [45–47] obtains Pareto front approximations via alternating optimizing one of the objectives. This approach is also limited to the treatment of bi-objective problems.

In [48], a continuation method is presented that is applicable to box-constrained BOPs. In [49], a variant of the method of Hillermeier is presented that is designed for the treatment of high-dimensional problems.

Recently, the Pareto Tracer (PT) was proposed by Martin and Schütze [2]. Similar to the method of Hillermeier, PT addresses the underdetermined nonlinear system of equations that is induced by the KKT equations. However, unlike the method of Hillermeier, the PT aims to separate the decision variables from the associated weight (or Lagrange) vectors whenever possible, leading to significant changes. The latter is due to the fact that the nonlinearity of the equation system can be significantly higher in the compound space compared to the corresponding system that is only defined in decision variable space. As a by-product, the chosen approach allows to compute the tangent space of both

Pareto set and front at every given regular point x . In [50], elements of the PT are used to treat many objective optimization problems (i.e., MOPs with more than, e.g., five objectives). Thus far, PT is only applicable to box and equality constrained problems which limits its application. In the following, we propose and discuss an extension of this method to adequately treat general MOPs, i.e., MOPs that in particular contain general inequalities.

3. Adapting the Pareto Tracer for General Inequality Constrained MOPs

In this section, we adapt the PT so that it can handle general inequality constraints. The core is the predictor–corrector step that generates from a given candidate solution x_i the following candidate x_{i+1} that satisfies the KKT conditions, and so that $F(x_{i+1}) - F(x_i)$ defines a pre-described movement in objective space along the set of KKT points.

Assume we are given a MOP of form (1) and a feasible point x_0 that satisfies the KKT conditions (3), where $\alpha_i > 0, i = 1, \dots, k$. Let $\epsilon > 0$ and define by

$$I_p(\epsilon) := \{j \in \{1, \dots, m\} : g_j(x_0) \geq -\epsilon\} \tag{4}$$

the set of indices corresponding to the nearly active inequalities at x_0 . If $I_p(\epsilon) = \{j_1, \dots, j_s\}, s \leq m$, define

$$G_\epsilon := \begin{pmatrix} \nabla g_{j_1}(x_0)^T \\ \vdots \\ \nabla g_{j_s}(x_0)^T \end{pmatrix} \in \mathbb{R}^{s \times n}. \tag{5}$$

Further, let

$$J := \begin{pmatrix} \nabla f_1(x)^T \\ \vdots \\ \nabla f_k(x)^T \end{pmatrix} \in \mathbb{R}^{k \times n} \tag{6}$$

$$H := \begin{pmatrix} \nabla h_1(x_0)^T \\ \vdots \\ \nabla h_p(x_0)^T \end{pmatrix} \in \mathbb{R}^{p \times n},$$

and $\alpha \in \mathbb{R}^n, \lambda \in \mathbb{R}^p$, and $\gamma \in \mathbb{R}^s$ be the solution of

$$\min_{\bar{\alpha}, \bar{\lambda}, \bar{\gamma}} \left\{ \left\| J^T \bar{\alpha} + H^T \bar{\lambda} + G_\epsilon^T \bar{\gamma} \right\|_2^2 : \bar{\alpha}_i \geq 0, i = 1, \dots, k, \sum_{i=1}^k \bar{\alpha}_i = 1 \right\}. \tag{7}$$

Note that (7) yields the Lagrange multipliers at x_0 for $\epsilon = 0$ if x_0 is a KKT point and if all active inequalities are regarded as equalities. Using α, λ and γ , define the matrix

$$W_{\alpha, \beta, \gamma} := \sum_{i=1}^k \alpha_i \nabla^2 f_i(x) + \sum_{i=1}^p \lambda_i \nabla^2 h_i(x) + \sum_{i=1}^s \gamma_i \nabla^2 g_{j_i}(x) \in \mathbb{R}^{n \times n}. \tag{8}$$

To compute a predictor direction $v_\mu \in \mathbb{R}^n$, we solve the system

$$\begin{pmatrix} W_{\alpha, \lambda, \gamma} & H^T & G_\epsilon^T \\ H & 0 & 0 \\ G_\epsilon & 0 & 0 \end{pmatrix} \begin{pmatrix} v_\mu \\ \zeta \\ \sigma \end{pmatrix} = \begin{pmatrix} -J^T \mu \\ 0 \\ 0 \end{pmatrix}. \tag{9}$$

Note that system (9) depends on $\mu \in \mathbb{R}^k$. Before we specify this vector, we first simplify (9). Denote by

$$A := \begin{pmatrix} H \\ G_\epsilon \end{pmatrix} \in \mathbb{R}^{(p+s) \times n}, \quad \xi := \begin{pmatrix} \zeta \\ \sigma \end{pmatrix} \in \mathbb{R}^{p+s}, \tag{10}$$

then (9) is equivalent to

$$\begin{pmatrix} W_{\alpha,\lambda,\gamma} & A^T \\ A & 0 \end{pmatrix} \begin{pmatrix} v_\mu \\ \xi \end{pmatrix} = \begin{pmatrix} -J^T \mu \\ 0 \end{pmatrix}. \tag{11}$$

Let $d \in \mathbb{R}^k$. It is straightforward to show that for a vector v_{μ_d} that solves (11), where $\mu_d \in \mathbb{R}^k$ is chosen such that

$$\begin{pmatrix} -JW_{\alpha,\lambda,\gamma}^{-1}J^T \\ 1 \dots 1 \end{pmatrix} \mu_d = \begin{pmatrix} d \\ 0 \end{pmatrix}, \tag{12}$$

it holds

$$Jv_{\mu_d} = d. \tag{13}$$

That is, (infinitesimal) small steps from x_0 into direction v_{μ_d} (in decision variable space) will lead to a movement from $F(x_0)$ into direction d (in objective space). It remains to select a suitable choice for d . Since α is orthogonal to the linearized Pareto front at $F(x_0)$ [1], a suggesting choice is hence by (13) to take d orthogonal to α . For this, let

$$\alpha = QR = (q_1, q_2, \dots, q_k)R, \tag{14}$$

where $Q \in \mathbb{R}^{k \times k}$ is orthogonal and $R \in \mathbb{R}^{k \times 1}$, be a QR-factorization of α . Then, any vector

$$d \in \text{span}\{q_2, \dots, q_k\} \tag{15}$$

can be chosen so that a movement in direction v_{μ_d} (in decision variable space) leads to a movement from $F(x_0)$ along the Pareto front. Note that the second equation in (12) reads as $\sum_{i=1}^k \mu_i = 0$. Hence, for the special case of a bi-objective optimization problem (i.e., $k = 2$), there are—after normalization—only two choices for μ :

$$\mu^{(1)} = \begin{pmatrix} -1 \\ 1 \end{pmatrix}, \quad \text{and} \quad \mu^{(2)} = \begin{pmatrix} 1 \\ -1 \end{pmatrix}. \tag{16}$$

Analog to Martin and Schütze [2], one can show that $\mu^{(1)}$ corresponds to a “right down” movement along the Pareto front while $\mu^{(2)}$ corresponds to a “left up” movement along the Pareto front.

After selecting the predictor direction v_μ , the question is how far to step in this direction. Here, we follow the suggestion made by Hillermeier [1] and use the step size

$$t = \frac{\tau}{\|Jv_\mu\|_2} \tag{17}$$

for a (small) value $\tau > 0$ so that

$$\|F(x_0 + tv_\mu) - F(x_0)\|_2 \approx \tau. \tag{18}$$

For the computations presented below, we make the following modifications: instead of $W_{\alpha,\beta,\gamma}$, we use the matrix

$$W_\alpha := \sum_{i=1}^k \alpha_i \nabla^2 f_i(x) \in \mathbb{R}^{n \times n}. \tag{19}$$

More precisely, for the computation of v_μ , we use the system

$$\begin{pmatrix} W_\alpha & A^T \\ A & 0 \end{pmatrix} \begin{pmatrix} v_\mu \\ \xi \end{pmatrix} = \begin{pmatrix} -J^T \mu \\ 0 \end{pmatrix} \tag{20}$$

and to obtain μ_d we solve

$$\begin{pmatrix} -JW_\alpha^{-1}J^T \\ 1 \dots 1 \end{pmatrix} \mu_d = \begin{pmatrix} d \\ 0 \end{pmatrix}. \tag{21}$$

We have observed similar performance for both approaches, while the usage of W_α compared to $W_{\alpha,\beta,\gamma}$ comes with the advantage that no Hessians for any of the constraint functions have to be computed.

Given a predictor point

$$\tilde{x}_1 := x_0 + t\nu_\mu \tag{22}$$

the task of the upcoming corrector step is to find a KKT point x_1 that is ideally near to \tilde{x}_1 . For this, we suggest to apply the multi-objective Newton method proposed in [51]. In particular, we first compute the solution $(\tilde{\nu}_1, \tilde{\delta})$ of the following problem

$$\begin{aligned} \min_{(\nu, \delta) \in \mathbb{R}^n \times \mathbb{R}} \quad & \delta \\ \text{s.t.} \quad & \nabla f_i(\tilde{x}_1)^T \nu + \frac{1}{2} \nu^T \nabla^2 f_i(\tilde{x}_1) \nu \leq \delta, \quad i = 1, \dots, k, \\ & h_i(\tilde{x}_1) + \nabla h_i(\tilde{x}_1)^T \nu = 0, \quad i = 1, \dots, p. \end{aligned} \tag{23}$$

$\tilde{\nu}_1$ is indeed the Newton direction for equality constrained MOPs as suggested in [2]. To adequately treat the involved inequalities, however, we propose to use the solution of the following problem:

$$\begin{aligned} \min_{(\nu, \delta) \in \mathbb{R}^n \times \mathbb{R}} \quad & \delta \\ \text{s.t.} \quad & \nabla f_i(\tilde{x}_1)^T \nu + \frac{1}{2} \nu^T \nabla^2 f_i(\tilde{x}_1) \nu \leq \delta, \quad i = 1, \dots, k, \\ & h_i(\tilde{x}_1) + \nabla h_i(\tilde{x}_1)^T \nu = 0, \quad i = 1, \dots, p, \\ & g_i(\tilde{x}_1) + \nabla g_i(\tilde{x}_1)^T \nu = 0, \quad i \in I_c(\epsilon). \end{aligned} \tag{24}$$

Note that problem (24) is identical to problem (23) except that $|I_c(\epsilon)|$ inequalities are treated as equalities at \tilde{x}_1 . In particular, we propose to add an index i to $I_c(\epsilon)$ if

- (a) $g_i(\tilde{x}_1) > \epsilon$, i.e., if \tilde{x}_1 significantly violates the constraint g_i ; or
- (b) $g_i(\tilde{x}_1) \in (-\epsilon, \epsilon)$ and $\nabla g(\tilde{x}_1)^T \tilde{\nu}_1 > 0$, i.e., if x_i is either active but g_i nearly active at x_i or if x_i already slightly violates g_i and a step into direction $\tilde{\nu}_1$ would lead to (further) violation of this constraint, indicated by $\nabla g(\tilde{x}_1)^T \tilde{\nu}_1 > 0$.

Algorithm 1 shows the pseudo code to build the index set $I_c(\epsilon)$ at a predictor point \tilde{x}_i . Given the Newton direction, the Newton step can then be performed via using the Armijo rule described in [51], as done in our computations. The set $I_c(\epsilon)$ is only computed once, it and remains fixed during the Newton iteration in the corrector step.

Algorithm 2 shows the pseudo code of one predictor–corrector step of the PT for general (equality and inequality constrained) MOPs. For bi-objective problems, μ can be chosen as in (16) leading either to a “left up” or “right down” movement, as discussed above. The algorithm has to be stopped if α is either close enough to $(1, 0)^T$ or $(0, 1)^T$, depending of course on the chosen search direction. For $k > 2$, one can use the box partition in objective space as described in [2] in order to mark the regions of the Pareto front that have already been “covered” during the run of the algorithm.

For the realization of the predictor–corrector step several linear systems of equations have to be solved, the largest one being (20). The cost is hence $O((n + p + s)^3)$ in terms of flops and $O((n + p + s)^2)$ in terms of storage. Further, for the corrector step the SOP (7) has to be solved that contains $k + p + s$ decision variables. For the computation of the Newton direction, the SOPs (23) and (24) have to be solved for the first Newton iteration that contains both $n + 1$ decision variables. For further Newton iterations, only SOP (24) has to be solved since the index set $I_c(\epsilon)$ remains fixed within a corrector step.

Finally, note that, if the method is realized as described above, the Hessians of all individual objectives have to be computed at each candidate solution (including at each Newton iteration). Using ideas from quasi-Newton methods, one can approximate the Hessians so that only gradient information is needed at each candidate solution, as described in [2].

Algorithm 1 Build $I_c(\epsilon)$

Require: \tilde{x}_i : predictor, \tilde{v}_i : corrector direction for (23), $\epsilon > 0$: tolerance

Ensure: $I_c(\epsilon)$: index set

```

1:  $I := \emptyset$ 
2: for  $i = 1, \dots, m$  do
3:   if  $g_i(\tilde{x}_i) > \epsilon$  then
4:      $I := I \cup i$ 
5:   else if  $g_i(\tilde{x}_i) \in (-\epsilon, \epsilon) \wedge \nabla g(\tilde{x}_i)^T \tilde{v}_i > 0$  then
6:      $I := I \cup i$ 
7:   end if
8: end for
9: Return  $I_c(\epsilon)$ 

```

Algorithm 2 Predictor–corrector step of the Pareto Tracer for general MOPs

Require: x_i : current candidate solution, $\tau > 0$: desired distance in objective space, $\epsilon > 0$: tolerance

Ensure: x_{i+1} : new candidate solution

```

1: Compute  $\alpha^{(i)} \in \mathbb{R}^k$  via solving (7)
2: Compute  $W_\alpha^{(i)}$  as in (19)
3: Compute  $A$  as in (10)
4: Select  $\mu^{(i)}$  as in (16) or via (15) and (21)
5: Compute  $v_\mu^{(i)}$  via solving (20)
6:  $t^{(i)} := \frac{\tau}{\|v_\mu^{(i)}\|_2}$ 
7:  $\tilde{x}_{i+1} := x_i + t^{(i)}v_\mu^{(i)}$ 
8: Compute  $x_{i+1}$  via a Newton method starting at  $\tilde{x}_{i+1}$ . For the Newton direction use the solution
   of (24).
9: Return  $x_{i+1}$ 

```

As a demonstration example, we consider the problem

$$\begin{aligned}
 & \min \begin{cases} f_1(x) = (x_1 + 3)^2 + (x_2 - 2)^2, \\ f_2(x) = x_1^2 + (x_2 + 3)^2, \end{cases} \\
 \text{s.t. } & g_1(x) = (x_1 + 1)^2 + x_2^2 \leq 2^2, \\
 & g_2(x) = (x_1 + 2)^2 + (x_2 + 2)^2 \leq 2^2.
 \end{aligned} \tag{25}$$

Figure 1a shows the Pareto set of the above problem where the two inequalities have been left out (i.e., the line segment connecting $(-3, 2)^T$ and $(0, -3)^T$), the sets $g_i(x) = 0$, $i = 1, 2$, as well as the Pareto set of this problem which is indeed the result of the PT. As starting point, we chose a point which significantly violates both constraints (and, hence, $|I_c(\epsilon)| = 2$ for $\epsilon = 1e - 4$). An application of the above-described Newton method leads to the point on the Pareto set with the smallest x_1 -value, which is in fact the initial point for the PT. During the run of PT, first only g_2 is “active” in the corrector step (i.e., $I_c(\epsilon) = \{2\}$), later none of the constraints (in the intersection of the Pareto fronts of the constrained and the unconstrained MOP), and finally only g_1 .

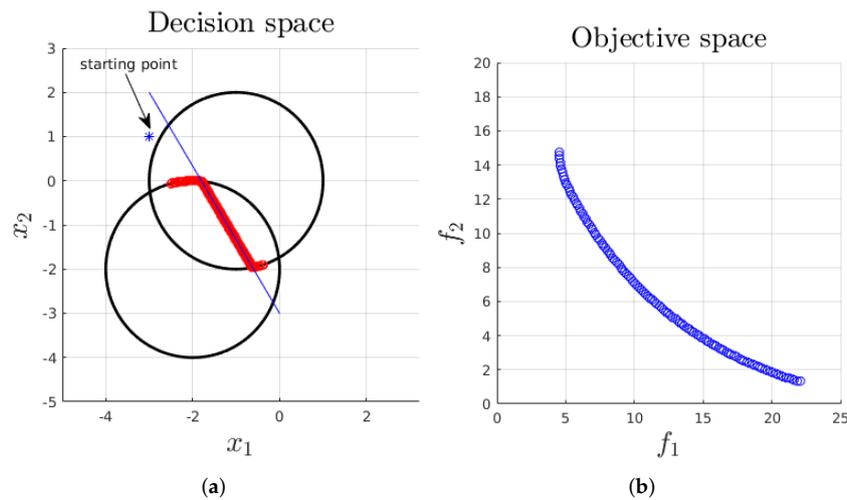


Figure 1. Numerical result of the PT for MOP (25).

4. Numerical Results

In this section, we further demonstrate the behavior of the PT on five benchmark problems that contain inequality constraints. For all problems, we used the quasi-Newton variant of PT that only required function and Jacobian information (and no Hessians). To compare the results, we also show the respective results obtained by the normal boundary intersection (NBI, [16]), the ϵ -constraint method [5], and the multi-objective evolutionary algorithm NSGA-II. For NBI and the ϵ -constraint method, we used the code that is available at [52], and for NSGA-II the implementation of PlatEMO [53]. Regrettably, no comparison to a multi-objective continuation method can be presented since none of the respective codes are publicly available. For a comparison of the PT and the method of Hillermeier on box and equality constrained MOPs, we refer to [2]. We chose also to include a comparison to the famous NSGA-II since it is widely used and state-of-the-art for two- and three-objective problems as we consider here. We stress that the comparisons only show (on the first four test problems) that PT outperforms NSGA-II on these particular cases where the Pareto front consists of one connected component. For highly multi-modal functions where the Pareto set/front falls into several connected components, NSGA-II will certainly outperform the (standalone) PT. A fair comparison can only be obtained when integrating PT into a global heuristic (as, e.g., done in [41]). This is certainly an interesting task, however, beyond the scope of this work.

To compare the results, we compare the total number of function evaluations used for each algorithm on each problem. For this, each Jacobian call is counted as four function calls assuming that the derivative is obtained via automatic differentiation [54]. To measure the quality of the approximations, we used the averaged Hausdorff distance Δ_2 [55–57]. Since NSGA-II has stochastic components, we applied this algorithm for each problem 10 times and present the median result (measured by Δ_2).

4.1. Binh and Korn

Our first test example is a modification of the box-constrained BOP from Binh and Korn [58], where we add two inequality constraints as follows:

$$\begin{aligned}
 & \min \begin{cases} f_1(x) = 4x_1^2 + 4x_2^2, \\ f_2(x) = (x_1 - 5)^2 + (x_2 - 5)^2, \end{cases} \\
 \text{s.t. } & (x_1 - 2)^2 + (x_2 - 1)^2 \leq 2.3^2, \\
 & (x_1 - 3)^2 + (x_2 - 3)^2 \geq 1.5^2, \\
 & 0 \leq x_1 \leq 5, \\
 & 0 \leq x_2 \leq 3.
 \end{aligned} \tag{26}$$

Table 1 shows the design parameters that have been used by NSGA-II for this problem, Table 2 shows the computational efforts and the obtained approximation quality for each algorithm, and Figures 2 and 3 show the obtained Pareto set and front approximations, respectively. For PT, we chose $\tau = 0.6$ leading to 52 solutions along the Pareto set/front in 4.48 s (the computations have been done on a Ubuntu 20.04.1 LTS system with an Intel Core i7-8550U 1.80 GHz x 8 CPU and 12 GB of RAM). We then applied NBI and the ϵ -constraint model using this number of sub-problems. For NSGA-II, we took the population size 100, which is a standard value for this algorithm. The results show nearly perfect Pareto front approximations (at least from the practical point of view) for all algorithms, which is also reflected by the low Δ_2 values that are very close to the optimal value of 0.6 (at least for PT, defined by τ). In terms of function evaluations, PT clearly wins over NBI and the ϵ -constraint method. A comparison to NSGA-II is not possible due to the choice of the population size.

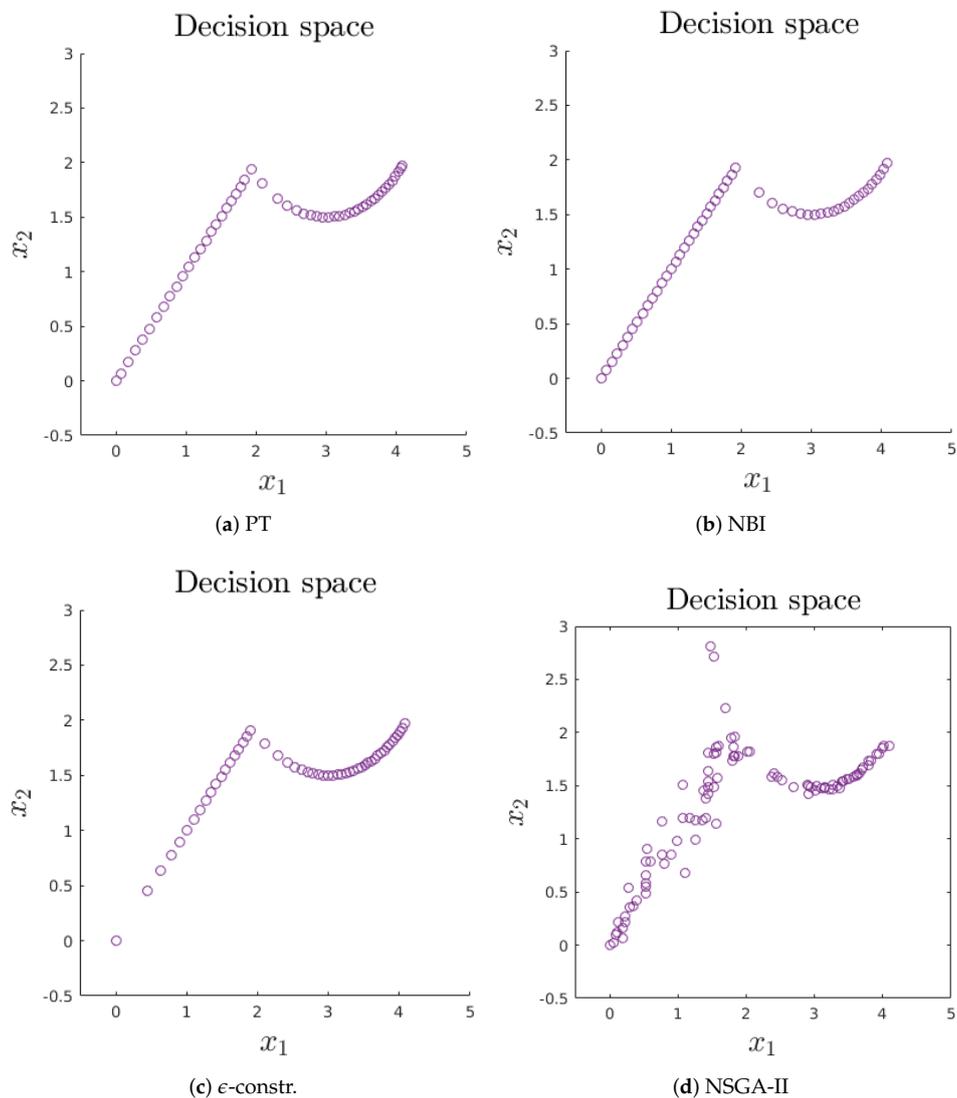


Figure 2. Results in decision space for MOP (26).

Table 1. Parameters used by NSGA-II for MOP (26).

Population Size	100
Number of generations	20
Probability of crossover	0.9
Probability of mutation	0.5

Table 2. Computational efforts and approximation quality of the algorithms for MOP (26).

	PT	NBI	ϵ -Constr.	NSGA-II
Solutions	52	52	52	100
Function Evaluations	151	427	336	2000
Jacobian Evaluations	133	425	336	-
Hessian Evaluations	-	373	284	-
Total of Evaluations	683	8095	6224	2000
Δ_2	0.6050	0.6025	0.9272	0.5626

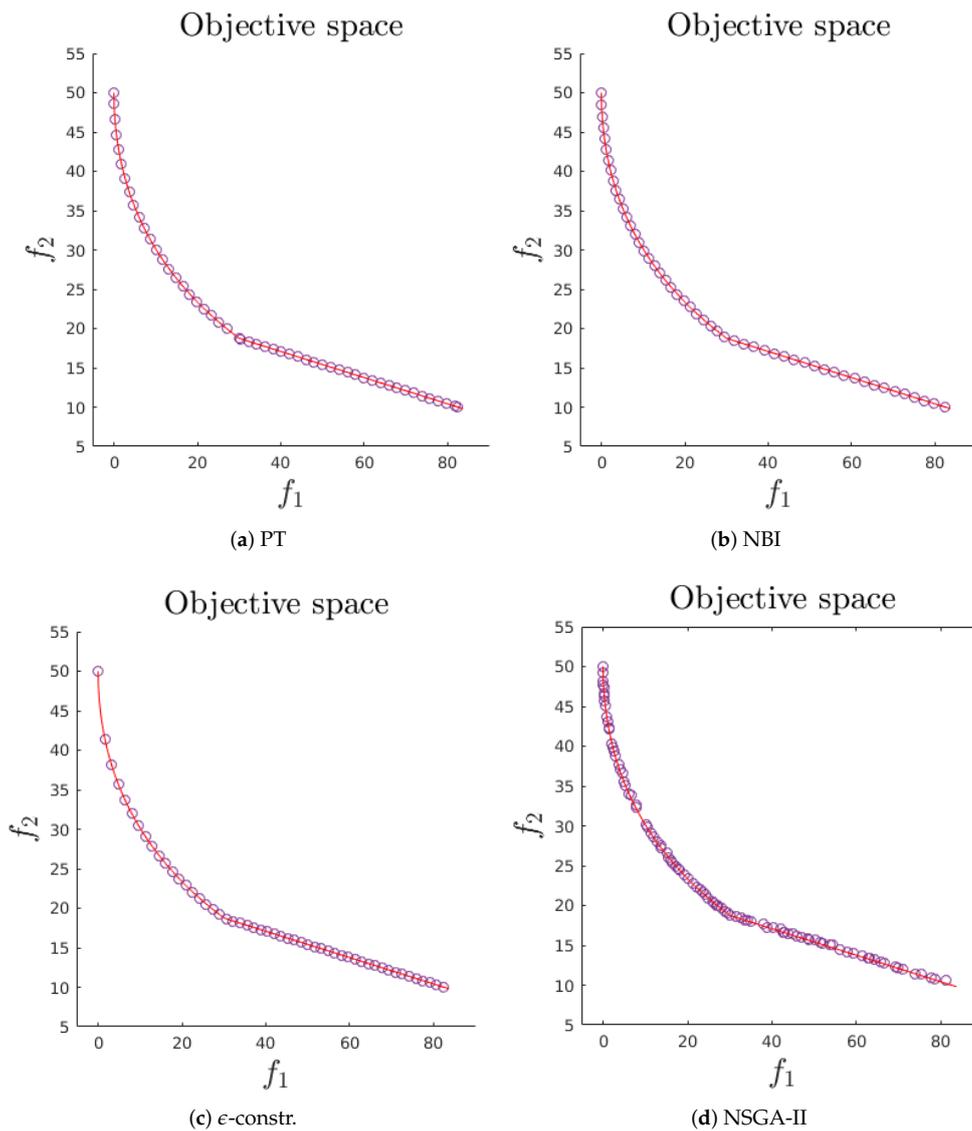


Figure 3. Results in objective space for MOP (26).

4.2. Chakong and Haimes

Next, we consider the bi-objective problem of Chakong and Haimes [59], which contains next to the box constraints one linear and one nonlinear inequality.

$$\begin{aligned}
 & \min \begin{cases} f_1(x) = 2 + (x_1 - 2)^2 + (x_2 - 1)^2, \\ f_2(x) = 9x_1 - (x_2 - 1)^2, \end{cases} \\
 & \text{s.t. } x_1^2 + x_2^2 \leq 225, \\
 & \quad x_1 - 3x_2 + 10 \leq 0, \\
 & \text{with } -20 \leq x_1, x_2 \leq 20.
 \end{aligned} \tag{27}$$

Table 3 shows the parameter values used for the application of NSGA-II, Table 4 the computational efforts and the approximation qualities, and Figures 4 and 5 the obtained approximations. We used $\tau = 1$ for PT, and proceeded as for the previous example for the other methods. The results are also similar to the previous example: all methods are capable of detecting a nearly perfect Pareto front approximation, and the overall cost is significantly less for PT, in 5.96 s.

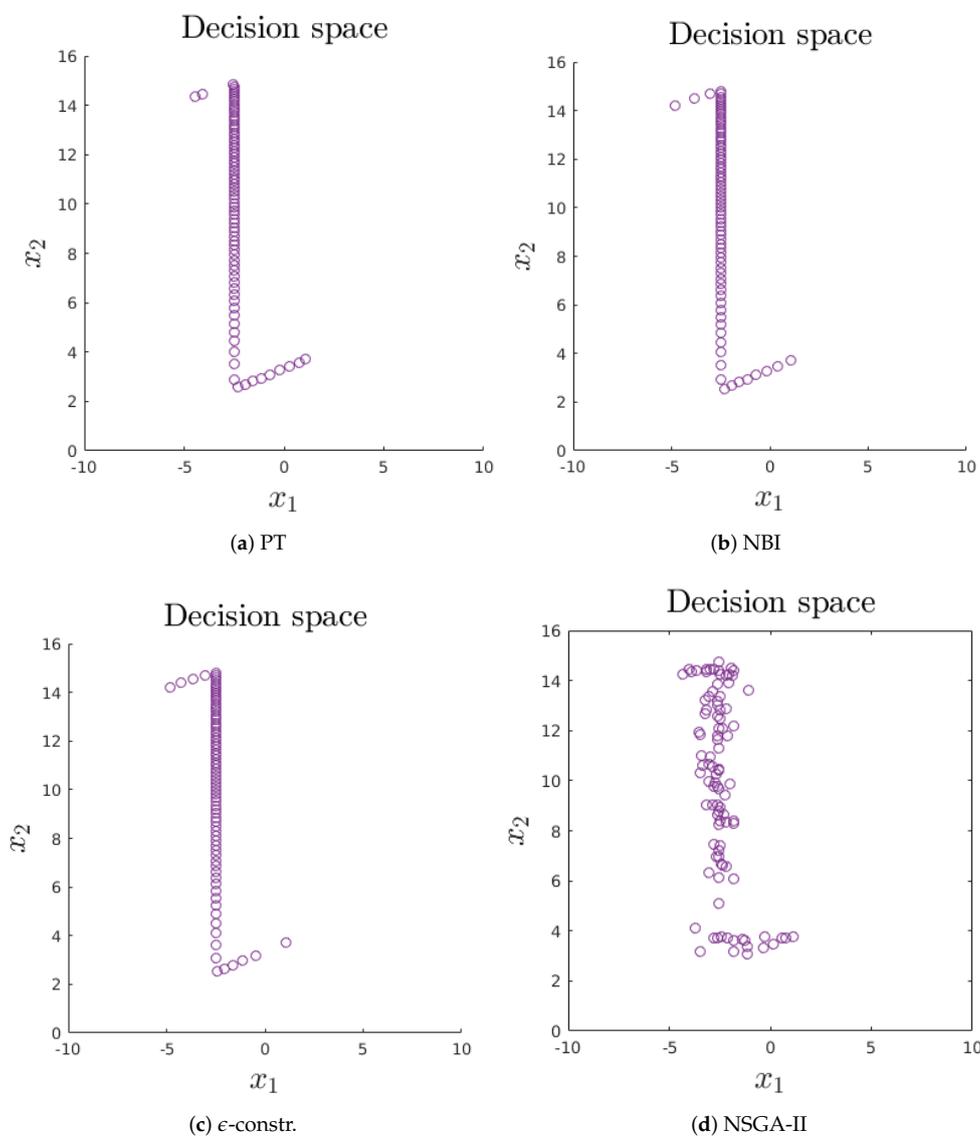


Figure 4. Results in decision space for MOP (27).

Table 3. Parameters used by NSGA-II for problem (27).

Population Size	100
Number of generations	30
Probability of crossover	0.9
probability of mutation	0.5

Table 4. Computational efforts and approximation qualities for problem (27).

	PT	NBI	ϵ -Constr.	NSGA-II
Solutions	80	80	80	100
Function Evaluations	540	678	578	3000
Jacobian Evaluations	499	678	578	-
Hessian Evaluations	-	598	498	-
Total of Evaluations	2536	12,958	10,858	3000
Δ_2	1.1459	1.1457	1.2141	1.1871

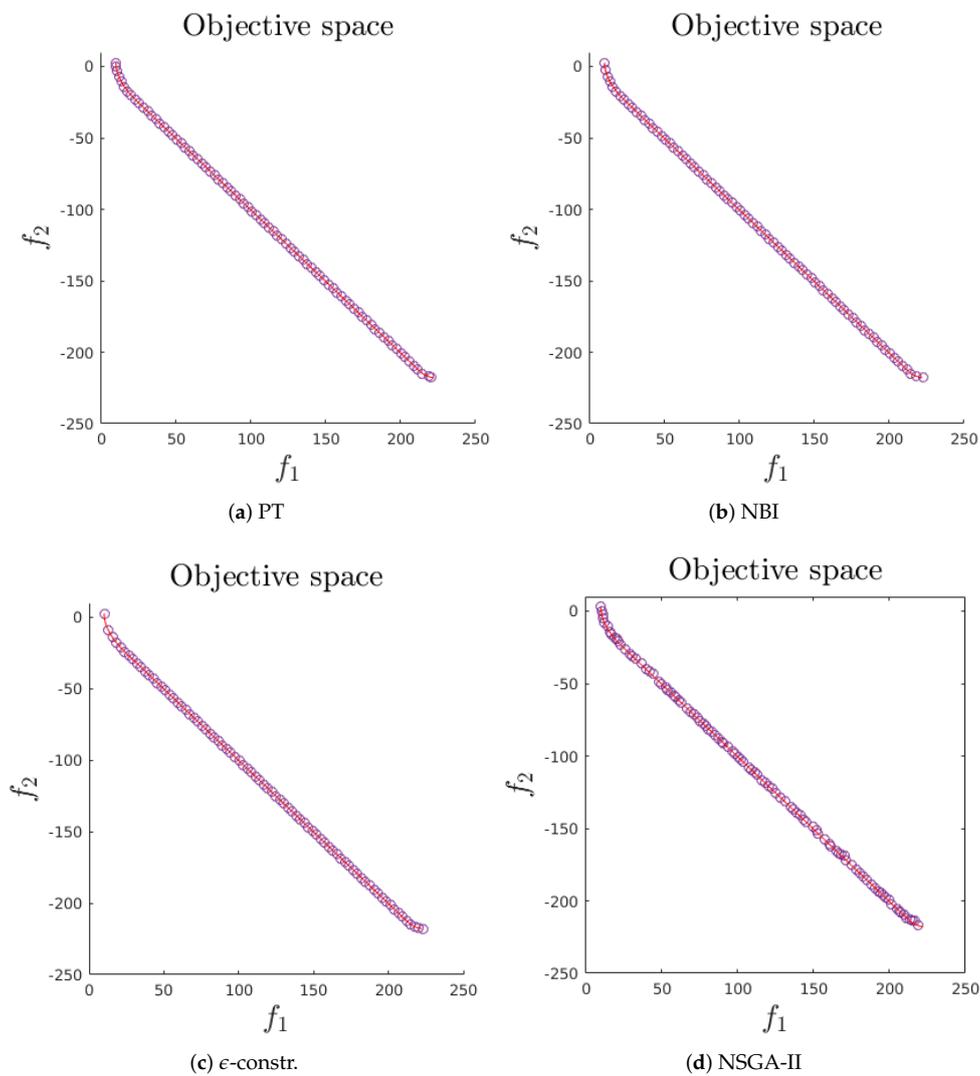


Figure 5. Results in objective space for MOP (27).

4.3. Tamaki

Next, we considered a MOP with three objectives (28):

$$\begin{aligned} \min \quad & \begin{cases} f_1(x) = x_1, \\ f_2(x) = x_2, \\ f_3(x) = x_3, \end{cases} \\ \text{s.t.} \quad & x_1^2 + x_2^2 + x_3^2 \geq 1, \\ & 0 \leq x_1, x_2, x_3 \leq 4. \end{aligned} \tag{28}$$

Both the Pareto set and front for this problem are a part of the unit sphere. Table 5 shows the design parameters for NSGA-II, Table 6 shows the computational effort and the approximation quality for each algorithm, and Figure 6 shows the Pareto front approximations (the respective Pareto set approximations will look identically, albeit in x -space). For this problem, $\tau = 0.05$ was used. The implementation of the ϵ -constrained method did not yield a result. On the Tamaki problem, PT performs better than the other algorithms both in approximation quality and in the overall computational cost.

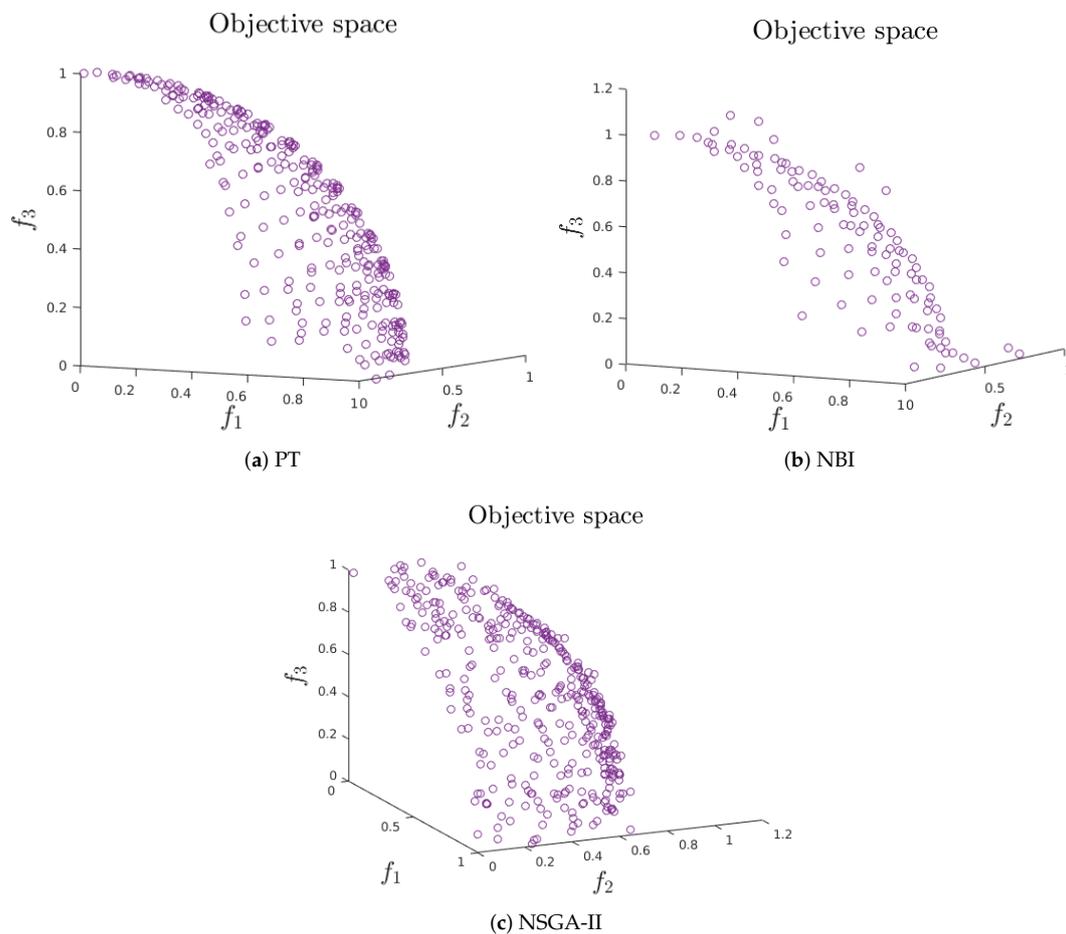


Figure 6. Results in objective space for MOP (28).

Table 5. Parameters used by NSGA-II for MOP (28).

Population Size	300
Number of generations	150
Probability of crossover	0.9
Probability of mutations	0.5

Table 6. Computational efforts and approximation qualities for problem (28).

	PT	NBI	ϵ -Constr.	NSGA-II
Solutions	305	112	N/A	300
Function Evaluations	2498	3758	N/A	450,000
Jacobian Evaluations	1101	3758	N/A	-
Hessian Evaluations	-	3293	N/A	-
Total of Evaluations	6902	91,236	N/A	450,000
Δ_2	0.0380	0.6353	N/A	0.0390

4.4. BCS

We next considered a second three-objective problem that contains next to one inequality also a linear equality constraint:

$$\begin{aligned}
 \min \begin{cases} f_1(x) = (x_1 + 3)^2 + (x_2 + 3)^2 + (x_3 + 3)^2, \\ f_2(x) = (x_1 - 9)^2 + (x_2 + 5)^2 + (x_3 + 5)^2, \\ f_3(x) = (x_1 - 5)^2 + (x_2 - 8)^2 + x_3^2, \end{cases} & (29) \\
 \text{s.t. } x_1 - 2x_2 - 3x_3 = 0, & \\
 \sin(2x_1) - x_2 \leq 0. &
 \end{aligned}$$

Table 7 presents the design parameters used by NSGA-II, Table 8 shows the computational effort and the approximation quality for each algorithm, and Figures 7 and 8 present the Pareto front approximation of PT (using $\tau = 2$), which took 16.79 s. For this example, none of the other methods were able to yield feasible solutions, where we counted a solution x to be feasible if $|x_1 - 2x_2 - 3x_3| < 1e - 4$ and $\sin(2x_1) - x_2 \leq 1e - 4$.

Decision space

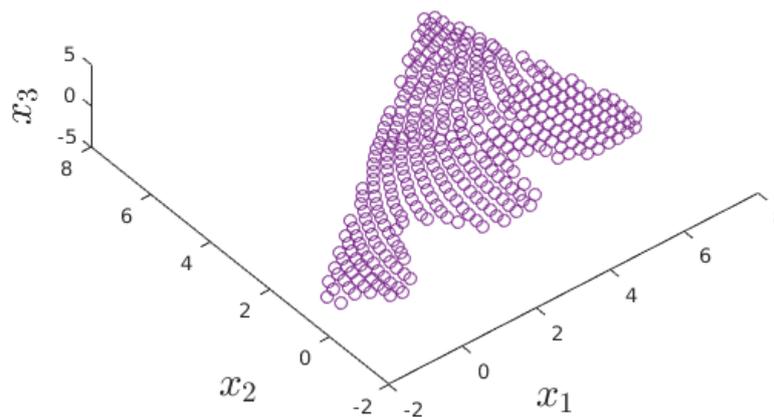


Figure 7. Numerical result of PT in the decision space for MOP (29).

Table 7. Parameters used by NSGA-II for MOP (29).

Population Size	100
Number of generations	500
Probability of crossover	0.9
Probability of mutations	0.5

Table 8. Computation efforts for the proposed test problem (29).

	PT	NBI	ϵ -Constr.	NSGA-II
Solutions	378	0	N/A	4
Function Evaluations	1923	2290	N/A	50,000
Jacobian Evaluations	756	1641	N/A	-
Hessian Evaluations	-	1431	N/A	-
Total of Evaluations	4947	40,336	N/A	50,000
Δ_2	2.0658	-	N/A	61.7685

Objective space

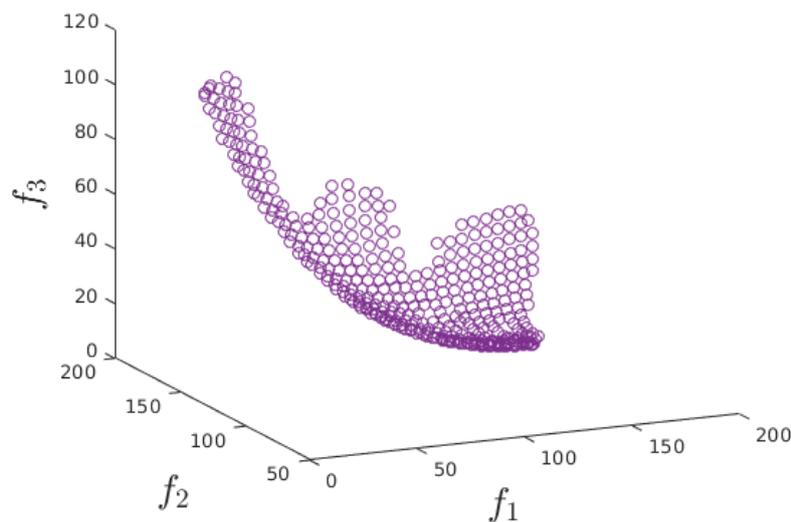


Figure 8. Numerical result of PT in the objective space for MOP (29).

4.5. Osyckza and Kundu

As last example, we considered the bi-objective problem of Osyckza and Kundu [60], which has six decision variables and contains six inequality constraints in addition to the box constraints:

$$\begin{aligned}
 \min \quad & \begin{cases} f_1(x) = -25(x_1 - 2)^2 - (x_2 - 2)^2 - (x_3 - 1)^2 - (x_4 - 4)^2 - (x_5 - 1)^2 \\ f_2(x) = \sum_{i=1}^6 x_i^2 \end{cases} \\
 \text{s.t.} \quad & x_1 + x_2 - 2 \geq 0 \\
 & 6 - x_1 - x_2 \geq 0 \\
 & 2 - x_2 + x_1 \geq 0 \\
 & 2 - x_1 + 3x_2 \geq 0 \\
 & 4 - (x_3 - 3)^2 - x_4 \geq 0 \\
 & (x_5 - 3)^2 + x_6 - 4 \geq 0 \\
 & 0 \leq x_1, x_2, x_6 \leq 10 \\
 & 1 \leq x_3, x_5 \leq 5 \\
 & 0 \leq x_4 \leq 6
 \end{aligned} \tag{30}$$

While the Pareto front of this problem is connected, its Pareto set consists of three different connected components. Hence, PT is not able to compute an approximation of the entire Pareto front with only one starting point. Figure 9a shows the result of PE for $\tau = 2$ using the three starting points

$$\begin{aligned} x_{0,1} &= (0.60, 1.50, 1.0, 0.00, 1.00, 0.04)^T, \\ x_{0,2} &= (0.00, 2.00, 2.20, 0.00, 1.00, 0.00)^T, \\ x_{0,3} &= (5.00, 1.00, 5.00, 0.00, 1.00, 0.01)^T. \end{aligned} \tag{31}$$

The computational time to obtain this result was 12.98 s. Figure 9b shows a numerical result of NSGA-II using the design parameters shown in Table 9. The obtained solutions “under” the Pareto front can be explained by the tolerance of 1×10^{-4} that was used to measure feasibility (while 1×10^{-8} was used for PT). Table 10 shows the computational effort for both methods. Needless to say, this represents by no means a comparison of the two methods. Instead, this should be rather seen as a motivation to hybridize PT with a global search strategy in order to obtain a fast and reliable multi-objective solver, which we leave for future studies.

Table 9. Parameters used by NSGA-II for MOP (30).

Population Size	435
Number of generations	50
Probability of crossover	0.9
Probability of mutations	0.5

Table 10. Computational efforts and approximation qualities for problem (30).

	PT	NSGA-II
Solutions	435	428
Function Evaluations	2051	20,000
Jacobian Evaluations	850	-
Hessian Evaluations	-	-
Total of Evaluations	5451	20,000
Δ_2	1.801	2.4819

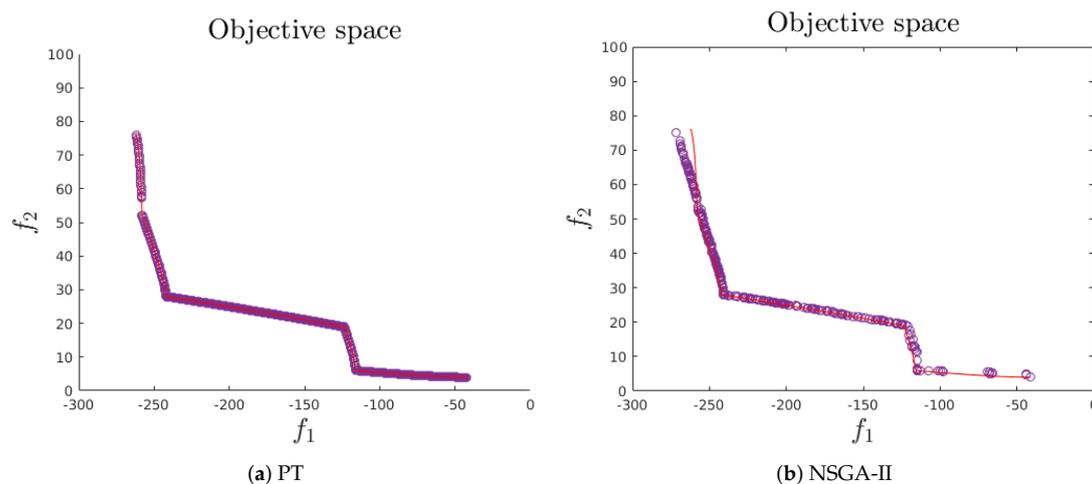


Figure 9. Results in objective space for MOP (30).

5. Conclusions and Future Work

In this paper, we extend the multi-objective continuation method Pareto Tracer (PT) for the treatment of general inequality constraints. To this end, the predictor–corrector step is modified

as follows: in the predictor, all nearly active inequalities are treated as equalities. In the following corrector step, the main challenge is to identify the inequalities for which the predictor solution is either nearly active or slightly violates the constraint that has to be considered, namely the equality constraint in the Newton method, and this is done in a bootstrap manner. We formulate the resulting algorithm and show some numerical results on several benchmark problems, indicating that it can reliably handle inequality (and equality) constrained MOPs. We further present comparisons to some other numerical methods. The results show that the extended PT can indeed reliably handle general MOPs (and in particular general inequalities). However, the method is—by construction—of local nature and restricted to the connected component of the solution set for which one initial solution is available. One interesting task is certainly to hybridize PT with a global solver such as a multi-objective evolutionary algorithm and to compare the resulting hybrid against other methods with respect to their ability to compute the entire global Pareto set/front of a given MOP. This is beyond the scope of this work and has been left for future work.

Author Contributions: Conceptualization and formal analysis, O.S.; software, F.B. and O.C.; and writing and editing: F.B., O.C., and O.S. All authors have read and agreed to the published version of the manuscript.

Funding: The authors acknowledge support from Conacyt project No. 285599 and SEP Cinvestav project No. 231.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Hillermeier, C. *Nonlinear Multiobjective Optimization: A Generalized Homotopy Approach*; Springer: Basel, Switzerland, 2001.
- Martín, A.; Schütze, O. Pareto Tracer: A predictor–corrector method for multi-objective optimization problems. *Eng. Optim.* **2018**, *50*, 516–536. [[CrossRef](#)]
- Deb, K. *Multi-Objective Optimization Using Evolutionary Algorithms*; John Wiley & Sons: Chichester, UK, 2001.
- Peitz, S.; Dellnitz, M. A Survey of Recent Trends in Multiobjective Optimal Control—Surrogate Models, Feedback Control and Objective Reduction. *Math. Comput. Appl.* **2018**, *23*, 30. [[CrossRef](#)]
- Miettinen, K. *Nonlinear Multiobjective Optimization*; Springer Science & Business Media: Boston, MA, USA, 2012.
- Ehrgott, M. *Multicriteria Optimization*; Springer: Berlin/Heidelberg, Germany, 2005.
- Karush, W. Minima of Functions of Several Variables with Inequalities as Side Constraints. Master’s Thesis, Department of Mathematics, University of Chicago, Chicago, IL, USA, 1939.
- Kuhn, H.W.; Tucker, A.W. Nonlinear programming. In *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, CA, USA, 31 July–12 August 1950*; University of California Press: Berkeley, CA, USA, 1951; pp. 481–492.
- Gass, S.; Saaty, T. The computational algorithm for the parametric objective function. *Nav. Res. Logist. Q.* **1955**, *2*, 39–45. [[CrossRef](#)]
- Mavrotas, G. Effective implementation of the ϵ -constraint method in Multi-Objective Mathematical Programming problems. *Appl. Math. Comput.* **2009**, *213*, 455–465.
- Steuer, R.E.; Choo, E.U. An Interactive Weighted Tchebycheff Procedure for Multiple Objective Programming. *Math. Program.* **1983**, *26*, 326–344. [[CrossRef](#)]
- Kim, J.; Kim, S.K. A CHIM-based interactive Tchebycheff procedure for multiple objective decision making. *Comput. Oper. Res.* **2006**, *33*, 1557–1574. [[CrossRef](#)]
- Wierzbicki, A.P. A mathematical basis for satisficing decision making. *Math. Model.* **1982**, *3*, 391–405. [[CrossRef](#)]
- Bogetoft, P.; Hallefjord, A.; Kok, M. On the convergence of reference point methods in multiobjective programming. *Eur. J. Oper. Res.* **1988**, *34*, 56–68. [[CrossRef](#)]
- Hernández Mejía, J.A.; Schütze, O.; Cuate, O.; Lara, A.; Deb, K. RDS-NSGA-II: A Memetic Algorithm for Reference Point Based Multi-objective Optimization. *Eng. Optim.* **2017**, *49*, 828–845. [[CrossRef](#)]
- Das, I.; Dennis, J.E. Normal-boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems. *SIAM J. Optim.* **1998**, *8*, 631–657. [[CrossRef](#)]

17. Klamroth, K.; Tind, J.; Wiecek, M. Unbiased Approximation in Multicriteria Optimization. *Math. Methods Oper. Res.* **2002**, *56*, 413–437. [[CrossRef](#)]
18. Fliege, J. Gap-free computation of Pareto-points by quadratic scalarizations. *Math. Methods Oper. Res.* **2004**, *59*, 69–89. [[CrossRef](#)]
19. Eichfelder, G. *Adaptive Scalarization Methods in Multiobjective Optimization*; Springer: Berlin/Heidelberg, Germany, 2008.
20. Hernández, C.; Naranjani, Y.; Sardahi, Y.; Liang, W.; Schütze, O.; Sun, J.Q. Simple Cell Mapping Method for Multi-objective Optimal Feedback Control Design. *Int. J. Dyn. Control* **2013**, *1*, 231–238. [[CrossRef](#)]
21. Xiong, F.R.; Qin, Z.C.; Xue, Y.; Schütze, O.; Ding, Q.; Sun, J.Q. Multi-objective optimal design of feedback controls for dynamical systems with hybrid simple cell mapping algorithm. *Commun. Nonlinear Sci. Numer. Simul.* **2014**, *19*, 1465–1473. [[CrossRef](#)]
22. Fernández, J.; Schütze, O.; Hernández, C.; Sun, J.Q.; Xiong, F.R. Parallel simple cell mapping for multi-objective optimization. *Eng. Optim.* **2016**, *48*, 1845–1868. [[CrossRef](#)]
23. Sun, J.Q.; Xiong, F.R.; Schütze, O.; Hernández, C. *Cell Mapping Methods—Algorithmic Approaches and Applications*; Springer: Singapore, 2019.
24. Schütze, O.; Mostaghim, S.; Dellnitz, M.; Teich, J. Covering Pareto Sets by Multilevel Evolutionary Subdivision Techniques. In *Proceedings of the International Conference on Evolutionary Multi-Criterion Optimization (EMO 2003), Faro, Portugal, 8–11 April 2003*; Fonseca, C.M., Fleming, P.J., Zitzler, E., Deb, K., Thiele, L., Eds.; Springer: Berlin/Heidelberg, Germany, 2003; pp. 118–132.
25. Dellnitz, M.; Schütze, O.; Hestermeyer, T. Covering Pareto Sets by Multilevel Subdivision Techniques. *J. Optim. Theory Appl.* **2005**, *124*, 113–155. [[CrossRef](#)]
26. Jahn, J. Multiobjective search algorithm with subdivision technique. *Comput. Optim. Appl.* **2006**, *35*, 161–175. [[CrossRef](#)]
27. Schütze, O.; Vasile, M.; Junge, O.; Dellnitz, M.; Izzo, D. Designing optimal low thrust gravity assist trajectories using space pruning and a multi-objective approach. *Eng. Optim.* **2009**, *41*, 155–181. [[CrossRef](#)]
28. Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* **2002**, *6*, 182–197. [[CrossRef](#)]
29. Coello Coello, C.A.; Lamont, G.B.; Van Veldhuizen, D.A. *Evolutionary Algorithms for Solving Multi-Objective Problems*, 2nd ed.; Springer: New York, NY, USA, 2007.
30. Beume, N.; Naujoks, B.; Emmerich, M. SMS-EMOA: Multiobjective selection based on dominated hypervolume. *Eur. J. Oper. Res.* **2007**, *181*, 1653–1669. [[CrossRef](#)]
31. Lara, A.; Sanchez, G.; Coello Coello, C.A.; Schütze, O. HCS: A new local search strategy for memetic multiobjective evolutionary algorithms. *IEEE Trans. Evol. Comput.* **2009**, *14*, 112–132. [[CrossRef](#)]
32. Zhou, A.; Qu, B.Y.; Li, H.; Zhao, S.Z.; Suganthan, P.N.; Zhang, Q. Multiobjective evolutionary algorithms: A survey of the state of the art. *Swarm Evol. Comput.* **2011**, *1*, 32–49. [[CrossRef](#)]
33. Cuate, O.; Schütze, O. Variation Rate to Maintain Diversity in Decision Space within Multi-Objective Evolutionary Algorithms. *Math. Comput. Appl.* **2019**, *24*, 82. [[CrossRef](#)]
34. Schütze, O.; Hernández, C. *Archiving Strategies for Evolutionary Multi-objective Optimization Algorithms*; Springer: Cham, Switzerland, 2021.
35. Harada, K.; Sakuma, J.; Kobayashi, S. Local search for multiobjective function optimization: Pareto descent method. In *Proceedings of the 8th Annual Conference on Genetic and Evolutionary Computation, Seattle, WA, USA, 8–12 July 2006*; pp. 659–666.
36. Schütze, O.; Coello, C.A.C.; Mostaghim, S.; Dellnitz, M.; Talbi, E.G. Hybridizing Evolutionary Strategies with Continuation Methods for Solving Multi-objective Problems. *IEEE Trans. Evol. Comput.* **2008**, *19*, 762–769. [[CrossRef](#)]
37. Zapotecas Martínez, S.; Coello Coello, C.A. A proposal to hybridize multi-objective evolutionary algorithms with non-gradient mathematical programming techniques. In *Proceedings of the 10th International Conference on Parallel Problem Solving From Nature (PPSN '08), Dortmund, Germany, 13–17 September 2008*; pp. 837–846.
38. Bosman, P.A.N. On gradients and hybrid evolutionary algorithms for real-valued multiobjective optimization. *IEEE Trans. Evol. Comput.* **2011**, *16*, 51–69. [[CrossRef](#)]
39. Schütze, O.; Martín, A.; Lara, A.; Alvarado, S.; Salinas, E.; Coello, C.A. The Directed Search Method for Multiobjective Memetic Algorithms. *J. Comput. Optim. Appl.* **2016**, *63*, 305–332. [[CrossRef](#)]

40. Schütze, O.; Alvarado, S.; Segura, C.; Landa, R. Gradient subspace approximation: A direct search method for memetic computing. *Soft Comput.* **2016**, *21*, 6331–6350. [[CrossRef](#)]
41. Cuate, O.; Ponsich, A.; Uribe, L.; Zapotecas, S.; Lara, A.; Schütze, O. A New Hybrid Evolutionary Algorithm for the Treatment of Equality Constrained MOPs. *Mathematics* **2020**, *8*, 7. [[CrossRef](#)]
42. Martin, B.; Goldsztejn, A.; Granvilliers, L.; Jermann, C. Certified Parallelotope Continuation for One-Manifolds. *SIAM J. Numer. Anal.* **2013**, *51*, 3373–3401. [[CrossRef](#)]
43. Martin, B.; Goldsztejn, A.; Granvilliers, L.; Jermann, C. On continuation methods for non-linear bi-objective optimization: Towards a certified interval-based approach. *J. Glob. Optim.* **2014**, *64*, 3–16. [[CrossRef](#)]
44. Pereyra, V.; Saunders, M.; Castillo, J. Equispaced Pareto front construction for constrained bi-objective optimization. *Math. Comput. Model.* **2013**, *57*, 2122–2131. [[CrossRef](#)]
45. Wang, H. Zigzag Search for Continuous Multiobjective Optimization. *Inf. J. Comput.* **2013**, *25*, 654–665. [[CrossRef](#)]
46. Wang, H. Direct zigzag search for discrete multi-objective optimization. *Comput. Oper. Res.* **2015**, *61*, 100–109. [[CrossRef](#)]
47. Zhang, Q.; Li, F.; Wang, H.; Xue, Y. Zigzag search for multi-objective optimization considering generation cost and emission. *Appl. Energy* **2019**, 255. [[CrossRef](#)]
48. Recchioni, M.C. A path following method for box-constrained multiobjective optimization with applications to goal programming problems. *Math. Methods Oper. Res.* **2003**, *58*, 69–85. [[CrossRef](#)]
49. Ringkamp, M.; Ober-Blöbaum, S.; Dellnitz, M.; Schütze, O. Handling high dimensional problems with multi-objective continuation methods via successive approximation of the tangent space. *Eng. Optim.* **2012**, *44*, 1117–1146. [[CrossRef](#)]
50. Schütze, O.; Cuate, O.; Martín, A.; Peitz, S.; Dellnitz, M. Pareto Explorer: A global/local exploration tool for many-objective optimization problems. *Eng. Optim.* **2020**, *52*, 832–855. [[CrossRef](#)]
51. Fliege, J.; Drummond, L.M.G.; Svaiter, B.F. Newton’s Method for Multiobjective Optimization. *SIAM J. Optim.* **2009**, *20*, 602–626. [[CrossRef](#)]
52. Julia. JuMP—Julia for Mathematical Optimization. Available online: <http://www.juliaopt.org/JuMP.jl/v0.14/> (accessed on 17 December 2020).
53. Tian, Y.; Cheng, R.; Zhang, X.; Jin, Y. PlatEMO: A MATLAB platform for evolutionary multi-objective optimization [educational forum]. *IEEE Comput. Intell. Mag.* **2017**, *12*, 73–87. [[CrossRef](#)]
54. Griewank, A.; Walther, A. *Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation*; SIAM: Philadelphia, PA, USA, 2008.
55. Schütze, O.; Esquivel, X.; Lara, A.; Coello, C.A.C. Using the Averaged Hausdorff Distance as a Performance Measure in Evolutionary Multi-Objective Optimization. *IEEE Trans. Evol. Comput.* **2012**, *16*, 504–522. [[CrossRef](#)]
56. Bogoya, J.M.; Vargas, A.; Cuate, O.; Schütze, O. A (p, q) -averaged Hausdorff distance for arbitrary measurable sets. *Math. Comput. Appl.* **2018**, *23*, 51. [[CrossRef](#)]
57. Bogoya, J.M.; Vargas, A.; Schütze, O. The Averaged Hausdorff Distances in Multi-Objective Optimization: A Review. *Mathematics* **2019**, *7*, 894. [[CrossRef](#)]
58. Binh, T.T.; Korn, U. MOBES: A multiobjective evolution strategy for constrained optimization problems. In Proceedings of The Third International Conference on Genetic Algorithms (Mendel 97), Brno, Czech Republic, 25–27 June 1997; Volume 25, pp. 176–182.
59. Chankong, V.; Haimes, Y. *Multiobjective Decision Making: Theory and Methodology*; Dover Publications: Mineola, NY, USA, 2008.
60. Osyczka, A.; Kundu, S. A new method to solve generalized multicriteria optimization problems using the simple genetic algorithm. *Struct. Optim.* **1995**, *10*, 94–99. [[CrossRef](#)]

Publisher’s Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).