Ensuring Smoothly Navigable Approximation Sets by Bézier Curve Parameterizations in Evolutionary Bi-objective Optimization

Stefanus C. Maree[1(✉)], Tanja Alderliesten[2], and Peter A. N. Bosman[1]

[1] Life Sciences and Health Research Group, Centrum Wiskunde and Informatica, Amsterdam, The Netherlands
{maree,peter.bosman}@cwi.nl

[2] Department of Radiation Oncology, Leiden University Medical Center, Leiden, The Netherlands
t.alderliesten@lumc.nl

Abstract. The aim of bi-objective optimization is to obtain an approximation set of (near) Pareto optimal solutions. A decision maker then navigates this set to select a final desired solution, often using a visualization of the approximation front. The front provides a navigational ordering of solutions to traverse, but this ordering does not necessarily map to a smooth trajectory through decision space. This forces the decision maker to inspect the decision variables of each solution individually, potentially making navigation of the approximation set unintuitive. In this work, we aim to improve approximation set navigability by enforcing a form of smoothness or continuity between solutions in terms of their decision variables. Imposing smoothness as a restriction upon common domination-based multi-objective evolutionary algorithms is not straightforward. Therefore, we use the recently introduced uncrowded hypervolume (UHV) to reformulate the multi-objective optimization problem as a single-objective problem in which parameterized approximation sets are directly optimized. We study here the case of parameterizing approximation sets as smooth Bézier curves in decision space. We approach the resulting single-objective problem with the gene-pool optimal mixing evolutionary algorithm (GOMEA), and we call the resulting algorithm BezEA. We analyze the behavior of BezEA and compare it to optimization of the UHV with GOMEA as well as the domination-based multi-objective GOMEA. We show that high-quality approximation sets can be obtained with BezEA, sometimes even outperforming the domination- and UHV-based algorithms, while smoothness of the navigation trajectory through decision space is guaranteed.

Keywords: Evolutionary algorithm · Multi-objective optimization · Hypervolume · Bézier curve estimation · Approximation set navigation

T. Bäck et al. (Eds.): PPSN 2020, LNCS 12270, pp. 215–228, 2020.
https://doi.org/10.1007/978-3-030-58115-2_15

1 Introduction

The aim of multi-objective optimization is to obtain a set of solutions that is as close as possible to the set of Pareto-optimal solutions, with different trade-offs between the objective functions. A decision maker can then navigate the obtained set, called the *approximation set*, to select a desired solution. The decision maker often incorporates external factors in the selection process that are not taken into account in the optimization objectives. An inspection of the decision variables of individual solutions is therefore required to determine their desirability. To guide the selection in bi-objective optimization, a visualization of the *approximation front* (i.e., the approximation set mapped to objective space) or trade-off curve can be used. The approximation front then intuitively implies a navigational order of solutions by traversing the front from one end to the other. However, solutions with similar objective values could still have completely different decision values. The decision values of all solutions then need to be inspected individually and carefully because they may not change predictably when the approximation front is traversed. This could make navigation of the approximation set unintuitive and uninsightful.

Population-based multi-objective evolutionary algorithms (MOEAs) have successfully been applied to real-world black-box optimization problems, for which the internal structure is unknown, or too complex to exploit efficiently by direct problem-specific design [6,8,22]. However, imposing a form of smoothness or continuity in terms of decision variables between solutions in the approximation set as a restriction upon the population of MOEAs is not straightforward. An underlying requirement to do so is that control over approximation sets as a whole is needed. However, typical dominance-based EAs use single-solution-based mechanics. Alternatively, multi-objective optimization problems can be formulated as a higher-dimensional single-objective optimization problem by using a quality indicator that assigns a fitness value to approximation sets. An interesting quality indicator is the hypervolume measure [23], as it is currently the only known Pareto-compliant indicator, meaning that an approximation set of given size with optimal hypervolume is a subset of the Pareto set [9,13,24]. However, the hypervolume measure has large drawbacks when used as quality indicator in indicator-based optimization, as it does not take dominated solutions into account. The uncrowded distance has been recently introduced to overcome this [20], which then resulted in the uncrowded hypervolume (UHV) measure [18]. The UHV can be used directly as a quality indicator for indicator-based multi-objective optimization. To be able to optimize approximation sets in this approach, fixed-size approximation sets are parameterized by concatenating the decision variables of a fixed number of solutions [2,18,21]. A single-objective optimizer can then be used to directly optimize approximation sets. The resulting single-objective optimization problem is however rather high-dimensional. To efficiently solve it, the UHV gene-pool optimal mixing evolutionary algorithm (UHV-GOMEA) [18], exploits grey-box properties of the UHV problem by only updating a subset of the decision variables corresponding to one (or a few) multi-objective solutions.

In this work, we go beyond an unrestricted concatenation of the decision variables of solutions and we propose to model approximation sets as sets of points that lie on a Bézier curve [10] in decision space. Optimizing only the control points of the Bézier curve, that define its curvature, enforces the decision variables of solutions in the approximation set to vary in a smooth, continuous fashion, thereby likely improving intuitive navigability of the approximation set. Previous work on parameterizations of the approximation set has been applied mainly in a post-processing step after optimization, or was performed in the objective space [3,15,19], but this does not aid in the navigability of the approximation set in decision space. Moreover, fitting a smooth curve through an already optimized set of solutions might result in a bad fit, resulting in a lower-quality approximation set. Additionally, we will show that specifying solutions as points on a Bézier curve directly enforces a form of diversity within the approximation set, which can actually aid in the optimization process, and furthermore reduces the problem dimensionality of the single-objective problem.

The remainder of this paper is organized as follows. In Sect. 2, we introduce preliminaries on UHV-based multi-objective optimization. In Sect. 3, we define a measure for navigational smoothness of approximation sets. In Sect. 4, we introduce Bézier curves and the corresponding optimization problem formulation. Empirical benchmarking on a set of benchmark problems is performed in Sect. 5. Finally, we discuss the results and conclude in Sect. 6.

2 UHV-Based Multi-objective Optimization

Let $\mathbf{f} : \mathcal{X} \to \mathbb{R}^m$ be a to-be-minimized m-dimensional vector function and $\mathcal{X} \subseteq \mathbb{R}^n$ be the n-dimensional (box-constrained) decision space. When the objectives in $\mathbf{f}$ are conflicting, no single optimal solution exists, but the optimum of $\mathbf{f}$ can be defined in terms of *Pareto optimality* [14]. A solution $\mathbf{x} \in \mathcal{X}$ is said to *weakly dominate* another solution $\mathbf{y} \in \mathcal{X}$, written as $\mathbf{x} \preceq \mathbf{y}$, if and only if $f_i(\mathbf{x}) \leq f_i(\mathbf{y})$ for all i. When the latter relation is furthermore strict (i.e., $f_i(\mathbf{x}) < f_i(\mathbf{y})$) for at least one i, we say that $\mathbf{x}$ *dominates* $\mathbf{y}$, written as $\mathbf{x} \prec \mathbf{y}$. A solution that is not dominated by any other solution in $\mathcal{X}$ is called *Pareto optimal*. The *Pareto set* $\mathcal{A}^\star$ is the set of all Pareto optimal solutions, i.e., $\mathcal{A}^\star = \{\mathbf{x} \in \mathcal{X} : \nexists \mathbf{y} \in \mathcal{X} : \mathbf{y} \prec \mathbf{x}\} \subset \mathcal{X}$. The image of the Pareto set under $\mathbf{f}$ is called the *Pareto*

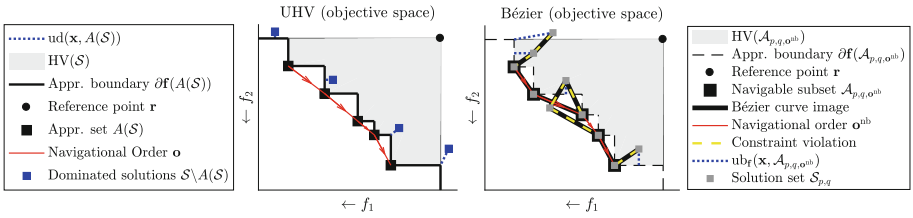

Fig. 1. Illustration of the uncrowded hypervolume (UHV) [18] (left) for a bi-objective minimization problem, and the Bézier parameterization (right).

front, i.e., $\{\mathbf{f}(\mathbf{x}) : \mathbf{x} \in \mathcal{A}^\star\} \subset \mathbb{R}^m$. The aim of multi-objective optimization is to approximate the Pareto set with a set of non-dominated solutions called an *approximation set* $\mathcal{A}$. Let $\mathcal{S} \subseteq \mathcal{X}$ be a solution set, that can contain dominated solutions and let $A : \wp(\mathcal{X}) \to \wp(\mathcal{X})$ be the approximation set given by $\mathcal{S}$, i.e., $A(\mathcal{S}) = \{\mathbf{x} \in \mathcal{S} : \nexists \mathbf{y} \in \mathcal{S} : \mathbf{y} \prec \mathbf{x}\}$, where $\wp(\mathcal{X})$ is the powerset of $\mathcal{X}$.

The hypervolume measure $\text{HV} : \wp(\mathcal{X}) \to \mathbb{R}$ [1,24] measures the area or volume dominated by all solutions in the approximation set, bounded by a user-defined reference point $\mathbf{r} \in \mathbb{R}^m$, as shown in Fig. 1. As the hypervolume ignores dominated solutions, we use the *uncrowded distance* to assign a quality value to dominated solutions [20]. The uncrowded distance $\text{ud}_\mathbf{f}(\mathbf{x}, \mathcal{A})$ measures the shortest Euclidean distance between $\mathbf{x}$ and the *approximation boundary* $\partial\mathbf{f}(\mathcal{A})$, when $\mathbf{x}$ is dominated by any solution in $\mathcal{A}$ or outside the region defined by $\mathbf{r}$, and is defined $\text{ud}_\mathbf{f}(\mathbf{x}, \mathcal{A}) = 0$ else (Fig. 1). It is called the uncrowded distance as the shortest distance to $\partial\mathbf{f}(\mathcal{A})$ is obtained for a point on the boundary that is not in $\mathcal{A}$ itself. Combining the uncrowded distance with the hypervolume measure results in the *uncrowded hypervolume* (UHV) [18],

$$\text{UHV}_\mathbf{f}(\mathcal{S}) = \text{HV}_\mathbf{f}(\mathcal{S}) - \frac{1}{|\mathcal{S}|} \sum_{\mathbf{x} \in \mathcal{S}} \text{ud}_\mathbf{f}(\mathbf{x}, A(\mathcal{S}))^m. \tag{1}$$

We use the subscript $\mathbf{f}$ to denote that its value is computed with respect to the multi-objective problem $\mathbf{f}$. To be able to optimize the UHV of a solution set, a parameterization of solution sets is required. Let $\phi \in \mathbb{R}^l$ be such a parameterization consisting of l decision variables, and let $S(\phi) = \{\mathbf{x}_1, \mathbf{x}_2, \ldots\}$ be an operator that transforms ϕ into its corresponding solution set. The resulting UHV-based optimization problem is then given by,

$$\begin{aligned} \text{maximize} \quad & \text{UHV}_{\mathbf{f},S}(\phi) = \text{HV}_\mathbf{f}(S(\phi)) - \frac{1}{|\mathcal{S}(\phi)|} \sum_{\mathbf{x} \in \mathcal{S}(\phi)} \text{ud}_\mathbf{f}(\mathbf{x}, A(\mathcal{S}(\phi)))^m, \\ \text{with} \quad & \mathbf{f} : \mathcal{X} \subseteq \mathbb{R}^n \to \mathbb{R}^m, \quad S : \mathbb{R}^l \to \wp(\mathcal{X}), \quad \phi \in \mathbb{R}^l. \end{aligned} \tag{2}$$

In a parameterization that is commonly used, solution sets $\mathcal{S}_p$ of fixed size p are considered, and the decision variables of the solutions in $\mathcal{S}_p$ are simply concatenated, i.e., $\phi = [\mathbf{x}_1 \cdots \mathbf{x}_p] \in \mathbb{R}^{p \cdot n}$ [2,18,21]. Using this parameterization, the resulting single-objective optimization problem is $l = p \cdot n$ dimensional. In [18], GOMEA [5] was used to efficiently solve this problem by exploiting the *grey-box* (gb) property that not all solutions $\mathbf{x}_i$ have to be recomputed when only some decision variables change. The resulting algorithm, which we call UHVEA-gb here (and was called UHV-GOMEA-Lm in [18]), greatly outperformed the mostly similar algorithm UHVEA-bb (called UHV-GOMEA-Lf in [18]) but in which the UHV was considered to be a *black box* (bb). This problem parameterization however does not guarantee any degree of navigational smoothness of the approximation set, which is the key goal in this paper.

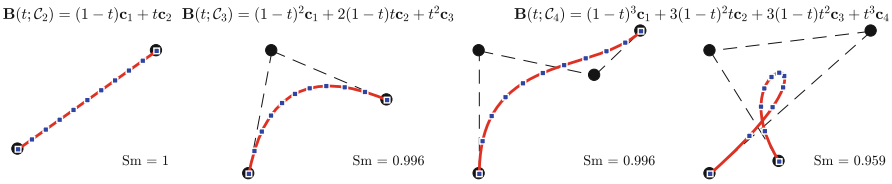

Fig. 2. Illustration of Bézier curves (red) in decision space with different control points (black). Blue points correspond to $p = 10$ evenly spread values of t, and the smoothness (Sm) of these p points is given, computed based on $\mathbf{o}^{\text{bez}}$. (Color figure online)

3 A Measure for Navigational Smoothness

We introduce a measure for the navigational smoothness of an approximation set. Let $\mathcal{S}_p = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_p\}$ be an approximation set of size p. Furthermore, let the *navigation order* $\mathbf{o}$ be a permutation of (a subset of) $I = \{1, 2, \ldots, p\}$, representing the indices of the solutions in $\mathcal{S}_p$ that the decision maker assesses in the order the solutions are inspected. The *(navigational) smoothness* $\text{Sm}(\mathcal{S}_p, \mathbf{o})$ is then defined as,

$$\text{Sm}(\mathcal{S}_p, \mathbf{o}) = \frac{1}{p-2} \sum_{i=2}^{p-1} \frac{\|\mathbf{x}_{\mathbf{o}_{i-1}} - \mathbf{x}_{\mathbf{o}_{i+1}}\|}{\|\mathbf{x}_{\mathbf{o}_{i-1}} - \mathbf{x}_{\mathbf{o}_i}\| + \|\mathbf{x}_{\mathbf{o}_i} - \mathbf{x}_{\mathbf{o}_{i+1}}\|}. \tag{3}$$

This smoothness measure measures the *detour length*, i.e., the extra distance traveled (in decision space) when going to another solution via an intermediate solution, compared to directly going there.

Throughout this work, we will consider a navigational order $\mathbf{o}$ for approximation sets $\mathcal{A}$ such that $f_1(\mathbf{x}_{\mathbf{o}_i}) < f_1(\mathbf{x}_{\mathbf{o}_j})$ holds whenever $i < j$ holds, i.e., from left to right in the objective space plot Fig. 1. We therefore simply write $\text{Sm}(\mathcal{A}, \mathbf{o}) = \text{Sm}(\mathcal{A})$ from now on. Note that $\text{Sm}(\mathcal{A}) \in [0, 1]$, and only if all solutions are colinear in decision space, $\text{Sm}(\mathcal{A}) = 1$ holds. This we consider the ideal scenario, where the decision variables of solutions change perfectly predictably. This also implies that any other (continuous) non-linear curve is not considered to be perfectly smooth. Although one could argue for different definitions of smoothness, we will see later that this measure serves our purpose for distinguishing smoothly from non-smoothly navigable approximation sets.

4 Bézier Curve Parameterizations of Approximation Sets

A Bézier curve $\mathbf{B}(t; \mathcal{C}_q)$ is a parametric curve that is commonly used in computer graphics and animations to model smooth curves and trajectories [10]. An n-dimensional Bézier curve is fully specified by an ordered set of $q \geq 2$ control points $\mathcal{C}_q = \{\mathbf{c}_1, \ldots, \mathbf{c}_q\}$ with $\mathbf{c}_j \in \mathbb{R}^n$, and given by,

$$\mathbf{B}(t; \mathcal{C}_q) = \sum_{j=1}^{q} b_{j-1,q-1}(t)\mathbf{c}_j, \quad \text{with} \quad b_{j,q}(t) := \binom{q}{j}(1-t)^{q-j}t^j, \tag{4}$$

for $0 \leq t \leq 1$, where $\binom{q}{j}$ are the binomial coefficients. Examples of Bézier curves are shown in Fig. 2. The first and last control points are always the end points of the Bézier curve, while intermediate control points do not generally lie on the curve. We parameterize a solution set $\mathcal{S}_p = \{\mathbf{x}_1, \ldots, \mathbf{x}_p\}$ of fixed size p using an n-dimensional Bézier curve $\mathbf{B}(t; \mathcal{C}_q)$ with q control points. On this curve, p points $\mathbf{x}_i = \mathbf{B}\left((i-1)/(p-1); \mathcal{C}_q\right)$ are selected, evenly spread in the domain of t. The resulting solution set $S_{p,q}(\phi) = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_p\}$ is then given by,

$$S_{p,q}(\phi) = \left\{ \mathbf{B}\left(\frac{0}{p-1}; \mathcal{C}_q\right), \mathbf{B}\left(\frac{1}{p-1}; \mathcal{C}_q\right), \ldots, \mathbf{B}\left(\frac{p-1}{p-1}; \mathcal{C}_q\right) \right\},$$

with $\phi = [\mathbf{c}_1 \cdots \mathbf{c}_q] \in \mathbb{R}^{q \cdot n}$. Note that inverting the order of control points does not affect the Bézier curve. To avoid this symmetry in the parameterization, we standardize the curve direction throughout optimization. After a change of the curve, we check if $f_1(\mathbf{c}_1) < f_1(\mathbf{c}_q)$ holds. If not, the order of the control points is simply inverted.

Algorithm 1: Navigational order for Bézier parameterizations

function: $[\mathcal{A}_{p,q,\mathbf{o}^{\text{nb}}}, (\mathbf{o}^{\text{nb}})] = \text{A}^{\text{nb}}(\mathcal{S}_{p,q}, \mathbf{o}^{\text{bez}})$
input : Bézier solution set $\mathcal{S}_{p,q} = \{\mathbf{x}_1, \ldots, \mathbf{x}_p\}$ with intrinsic ordering $\mathbf{o}^{\text{bez}}$
output : Approximation (sub)set $\mathcal{A}_{p,q,\mathbf{o}^{\text{nb}}}$, (navigational order $\mathbf{o}^{\text{nb}}$),

$\eta = \arg\min_{i \in \{1,\ldots,p\}} f_1(\mathbf{x}_{\mathbf{o}_i^{\text{bez}}})$;
$\mathbf{o}^{\text{nb}} = [\mathbf{o}_\eta^{\text{bez}}]$ and $\mathcal{A}_{p,q,\mathbf{o}^{\text{nb}}} = \{\mathbf{x}_{\mathbf{o}_\eta^{\text{bez}}}\}$;
for $j = \eta, \ldots, p$ **do**
  if $\mathbf{x}_{\mathbf{o}_j^{\text{bez}}} \in A(\mathcal{S}_{p,q})$ **and** $f_2(\mathbf{x}_{\mathbf{o}_j^{\text{bez}}}) < f_2(\mathbf{x}_{\mathbf{o}_{\text{end}}^{\text{nb}}})$ **then**
    $\mathbf{o}^{\text{nb}} = [\mathbf{o}^{\text{nb}}\ ;\ \mathbf{o}_j^{\text{bez}}]$ and $\mathcal{A}_{p,q,\mathbf{o}^{\text{nb}}} = \mathcal{A}_{p,q,\mathbf{o}^{\text{nb}}} \cup \{\mathbf{x}_{\mathbf{o}_j^{\text{bez}}}\}$; // here
    $\mathbf{o}_{\text{end}}^{\text{nb}} = \mathbf{o}_j^{\text{bez}}$

4.1 A Navigational Order for Bézier Parameterizations

Solution sets $\mathcal{S}_{p,q} = S_{p,q}(\phi)$ parameterized by a Bézier curve introduce an intrinsic order $\mathbf{o}^{\text{bez}}$ of solutions by following the curve from $t = 0$ to $t = 1$. Even though the solutions in $\mathcal{S}_{p,q}$ now lie on a smooth curve in decision space, it might very well be that some of these solutions dominate others. We define a navigational-Bézier (nb) order $\mathbf{o}^{\text{nb}}$ for a solution set $\mathcal{S}_{p,q}$ that follows the order of solutions $\mathbf{o}^{\text{bez}}$ along the Bézier curve, but also aligns with the left-to-right ordering described in Sect. 3. Pseudo code for $\mathbf{o}^{\text{nb}}$ is given in Algorithm 1, and an example is given in Fig. 1. The navigational order $\mathbf{o}^{\text{nb}}$ starts from the solution with best f_1-value and continues to follow the Bézier curve (i.e., in the order $\mathbf{o}^{\text{bez}}$) until the solution with best f_2-value is reached, only improving in f_2 (and thereby worsening in f_1) along the way, and skipping solutions that violate this

property. Let $\mathcal{A}_{p,q,\mathbf{o}^{\text{nb}}} = A^{\text{nb}}(\mathcal{S}_{p,q}, \mathbf{o}^{\text{bez}})$ be the resulting subset of $\mathcal{S}_{p,q}$ pertaining to exactly the solution indices as specified in $\mathbf{o}^{\text{nb}}$, and note that this is an approximation set.

4.2 Unfolding the Bézier Curve (in Objective Space)

Smoothly navigable approximation sets can now be obtained by maximizing the hypervolume of $\mathcal{A}_{p,q,\mathbf{o}^{\text{nb}}}$. To maximize the number of navigable solutions $|\mathcal{A}_{p,q,\mathbf{o}^{\text{nb}}}| = |\mathbf{o}^{\text{nb}}|$, we need to unfold the Bézier curve in objective space. For this, we introduce a constraint violation function $C(\mathcal{S}_{p,q}, \mathbf{o}^{\text{nb}}) \geq 0$, as given in Algorithm 2 and illustrated in Fig. 1. It is composed of two parts. The first part is similar to the uncrowded distance term in Eq. (1), but the approximation boundary is now given by $\mathcal{A}_{p,q,\mathbf{o}^{\text{nb}}}$. The second part aims to pull solutions that are not in $\mathcal{S}_{p,q,\mathbf{o}^{\text{nb}}}$ towards neighboring solutions on the Bézier curve.

Algorithm 2: Bézier constraint violation function

function: $C(\mathcal{S}_{p,q}, \mathbf{o}^{\text{bez}}) \geq 0$
input : Bézier solution set $\mathcal{S}_{p,q} = \{\mathbf{x}_1, \ldots, \mathbf{x}_p\}$ with intrinsic ordering $\mathbf{o}^{\text{bez}}$
output : Constraint value $C \geq 0$

$[\mathcal{A}, \mathbf{o}^{\text{nb}}] = \text{A}^{\text{nb}}(\mathcal{S}_{p,q}, \mathbf{o}^{\text{bez}})$; `// See Algorithm 1`
$C = \frac{1}{|\mathcal{S}_{p,q}|} \sum_{\mathbf{x} \in \mathcal{S}_{p,q}} \text{ud}_{\mathbf{f}}(\mathbf{x}, \mathcal{A}^m)$; `// Uncrowded distance (ud), see (1)`

for $j = 1, \ldots, |\mathcal{S}_{p,q}| - 1$ **do**
  if $\mathbf{o}^{\text{bez}}_j \notin \mathbf{o}^{nb}$ *or* $\mathbf{o}^{\text{bez}}_{j+1} \notin \mathbf{o}^{nb}$ **then**
    $C = C + \|\mathbf{f}(\mathbf{x}_{\mathbf{o}^{\text{bez}}_j}) - \mathbf{f}(\mathbf{x}_{\mathbf{o}^{\text{bez}}_{j+1}})\|$; `// Euclidean distance in` $\mathbb{R}^m$

4.3 Bézier Parameterization + GOMEA = BezEA

The resulting Bézier curve optimization problem is given by,

$$\begin{aligned} \text{maximize} \quad & \text{HV}_{\mathbf{f}, S_{p,q}}(\phi) = \text{HV}_{\mathbf{f}}(A^{\text{nb}}(S_{p,q}(\phi))), \\ \text{with} \quad & C(S_{p,q}(\phi), \mathbf{o}^{\text{nb}}(\phi)) = 0, \\ & \mathbf{f} : \mathcal{X} \subseteq \mathbb{R}^n \to \mathbb{R}^m, \quad S_{p,q} : \mathbb{R}^{q \cdot n} \to \wp(\mathcal{X}), \quad \phi \in \mathbb{R}^{q \cdot n}. \end{aligned} \tag{5}$$

We use *constraint domination* to handle constraint violations [7]. With constraint domination, the fitness of a solution is computed regardless of its feasibility. When comparing two solutions, if both are infeasible (i.e., $C > 0$), the solution with the smallest amount of constraint violation is preferred. If only one solution is infeasible, the solution that is feasible is preferred. Finally, if both solutions are feasible (i.e., $C = 0$), the original ranking based on fitness is used.

Bézier curves have no local control property, meaning that a change of a control point affects all solutions on the curve. Partial evaluations can therefore

no longer be exploited with this parameterization, and we thus solve this problem with the black-box version of GOMEA. Analogous to the UHV naming, we brand the resulting algorithm Bézier-GOMEA-bb, which we abbreviate to BezEA. A detailed description of GOMEA can be found in [5], and a description of UHV-GOMEA in [18].

5 Numerical Experiments

We compare BezEA with UHVEA-gb and UHVEA-bb. These methods use a different hypervolume-based representation of the multi-objective problem, but use very similar variation and selection mechanisms, making the comparison between these methods most fair. We use the guideline setting for the population size N of GOMEA with full linkage models in a black-box setting [4], which for separable problems yields $N = \lfloor 10\sqrt{l} \rfloor$ and for non-separable problems $N = 17 + \lfloor 3l^{1.5} \rfloor$. BezEA solves a single-objective problem of $l = qn$ decision variables. UHVEA-bb solves a single objective problem of $l = pn$ decision variables. UHVEA-gb solves the same problem by not considering all pn decision variables simultaneously, but by updating only subsets of $l = n$ decision variables, on which we base the population size guideline for UHVEA-gb.

We furthermore include the domination-based MO-GOMEA [6]. In MO-GOMEA, a population of solutions is aimed to approximate the Pareto front by implicitly balancing diversity and proximity. From a population of N_{mo} solutions, truncation selection is performed based on domination rank. The resulting selection is clustered into K_{mo} overlapping clusters that model different parts of the approximation front. For each cluster, a Gaussian distribution is estimated to sample new solutions from, which uses very similar update rules as the single-objective GOMEA, and therefore allows for a most fair comparison to BezEA and UHVEA. MO-GOMEA obtains an elitist archive, aimed to contain 1000 solutions. For a fair comparison to the hypervolume-based methods that obtain an approximation set of at most p solutions, we reduce the obtained elitist archive of MO-GOMEA to p solutions using greedy hypervolume subset selection (gHSS) [11], which we denote by MO-GOMEA*. As described in [18], to align MO-GOMEA with the other algorithms, we set $N_{mo} = p \cdot N$ and $K_{mo} = 2p$ such that the overall number of solutions in the populations is the same, and all sample distributions are estimated from the same number of solutions.

As performance measure, we define $\Delta\mathrm{HV}_p = \mathrm{HV}_p^\star - \mathrm{HV}(\mathcal{A}_p)$ as the distance to the optimal hypervolume $\mathrm{HV}_p^\star$ obtainable with p solutions, empirically determined with UHVEA.

5.1 Increasing q

We illustrate how increasing the number of control points q of the Bézier curve improves achievable accuracy of BezEA (with $q = \{2, \ldots, 10\}$ and $p = 10$) in case the Pareto set is non-linear. For this, we construct a simple two-dimensional problem *curvePS*, with objective functions $f_1^{\mathrm{curvePS}}(\mathbf{x}) = (x_1 - 1)^2 + 0.01x_2^2$ and

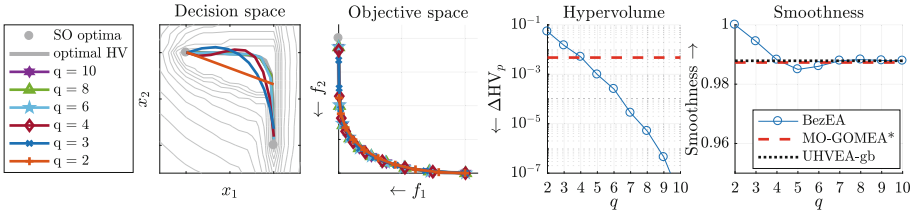

Fig. 3. Bézier curve approximations of the Pareto set of the curvePS problem (left), obtained with BezEA. Contour lines show domination ranks, the corresponding approximation fronts (middle), and ΔHV_{10} together with smoothness (right).

$f_2^{\text{curvePS}}(\mathbf{x}) = x_1^2 + (x_2 - 1)^2$. A large computational budget was used to show maximally achievable hypervolume, and standard deviations are therefore too small to be visible.

Results are shown in Fig. 3. A larger q results in a better approximation of the leftmost endpoint of the Pareto front (second subfigure), thereby improving ΔHV_p (third subfigure), but slightly lowering smoothness (fourth subfigure), as the Bézier curve deviates from a straight line. MO-GOMEA*, UHVEA-gb, and BezEA for large q all obtain a very similar smoothness. As MO-GOMEA* does not explicitly optimize the hypervolume of its approximation set, it obtains a slightly different distribution of solutions, which results in a lower hypervolume. Additionally, MO-GOMEA* does not converge to the Pareto set due to the finite population size and inifitely large Pareto set, as described in more detail in [18]. Even though this is a fundamental limitation of domination-based MOEAs, this level of accuracy is often acceptable in practice.

5.2 Comparison with UHV Optimization

Next, we demonstrate the behavior of BezEA compared to UHVEA on the simple *bi-sphere* problem, which is composed of two single-objective sphere problems, $f_{\text{sphere}}(\mathbf{x}) = \sum_{i=1}^{n} x_i^2$, of which one is translated, $f_1^{\text{bi-sphere}}(\mathbf{x}) = f_{\text{sphere}}(\mathbf{x})$, and $f_2^{\text{bi-sphere}} = f_{\text{sphere}}(\mathbf{x} - \mathbf{e}_1)$, where $\mathbf{e}_i$ is the i^{th} unit vector. We set $n = 10$, and initialize all algorithms in $[-5, 5]^n$. This is a separable problem and we therefore use the univariate population size guideline (i.e., $N = \sqrt{l}$). We consider the cases $p = \{10, 100\}$. The computational budget is set to $2p \cdot 10^4$ evaluations of the multi-objective problem given by $\mathbf{f}$ (MO-fevals). When the desired number of solutions p along the front is large, neighboring solutions are nearby each other on the approximation front. This introduces a dependency between these solutions, which needs to be taken into account in the optimization process to be able to effectively solve the problem [18].

Results are shown in Fig. 4. This problem is unimodal with a linear Pareto set, and the smoothness of (a subset) of the Pareto set is therefore 1.0. As UHVEA-gb converges to a subset of the Pareto set (see [18]), it ultimately obtains a smoothness of 1.0, even though its smoothness is initially lower. MO-GOMEA*

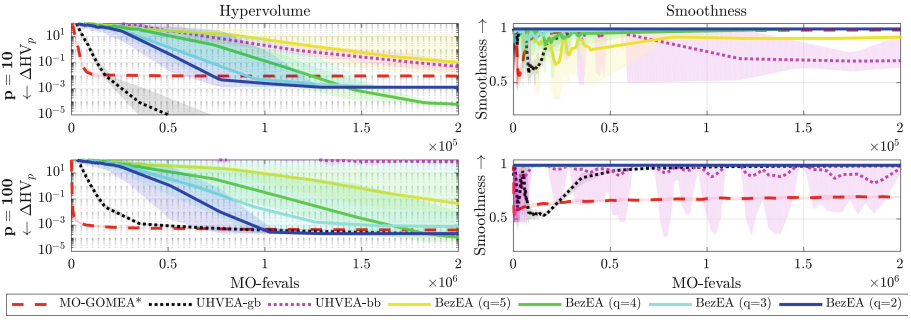

Fig. 4. Comparison of UHVEA with BezEA and MO-GOMEA* on the bi-sphere problem with $n = 10$ and $p = 10$ (top row) and $p = 100$ (bottom row). Left two subfigures show mean scores, and the shaded areas represent min/max scores, obtained over 10 runs. Objective and decision space subfigures show results of a single run. Solutions in the decision space projection are sorted based on their f_0-value, from best to worst.

does not converge to the Pareto set, and its smoothness stagnates close to 1.0 when $p = 10$, but stagnates around 0.7 when $p = 100$. BezEA with $q = 2$ has per construction a perfect smoothness of 1.0, and for $q = 3$ and $q = 4$, the obtained smoothness is close to 1. With $q = 5$ control points, BezEA does not converge within the given budget, resulting in a lower smoothness within the computational budget. UHVEA-gb furthermore shows a better convergence rate, which could be because UHVEA-gb can exploit partial evaluations, while this is not possible with BezEA. However, UHVEA-bb, which also does not perform partial evaluations, is unable to solve the problem for $p = 100$. This difference between BezEA and UHVEA-bb could be attributed to the lower degree of freedom that BezEA has due to the rather fixed distribution of solutions. This distribution does however not exactly correspond to the distribution of $\text{HV}_p^\star$. This is why a stagnation in terms of hypervolume convergence can be observed for small values of q. The solutions of BezEA are equidistantly distributed along the curve in terms of t. By doing so, intermediate control points can be used to adapt the distribution of solutions (when $q > 2$). This is why BezEA with $q = 4$ can obtain a better ΔHV_p than BezEA with $q = 2$, even though the Pareto set is linear. For $p = 100$, BezEA obtains a better ΔHV_p than UHVEA-gb, which can be explained by the increased problem complexity when the desired number of solutions along the front is large. Increasing the population size N of UHVEA-gb would (at least partially) overcome this, but we aimed here to show that BezEA does not suffer from this increased complexity as its problem dimensionality depends on q, not p.

5.3 WFG Benchmark

We benchmark BezEA, UHVEA, and MO-GOMEA on the nine commonly used WFG functions [12]. We consider bi-objective WFG problems with $n = 24$ decision variables of which $k_{\text{WFG}} = 4$ are WFG-position variables. We furthermore

Table 1. Obtained hypervolume HV_p (mean ± standard deviation (rank)) and mean navigational smoothness (Sm) for the 9 WFG problems with $p = 9$ solutions. Bold are best scores per problems, or those not statistically different from it.

#	MO-GOMEA*		UHVEA-gb		BezEA ($q = 2$)		BezEA ($q = 3$)	
	HV_9	Sm	HV_9	Sm	HV_9	Sm	HV_9	Sm
1	**97.60** ± 0.7 (1)	*0.76*	93.62 ± 1.7 (2)	*0.67*	90.35 ± 1.1 (4)	*1.00*	90.37 ± 1.2 (3)	*0.99*
2	110.09 ± 0.0 (2)	*0.86*	**110.38** ± 1.0 (1)	*0.66*	97.74 ± 0.0 (4)	*1.00*	97.85 ± 0.0 (3)	*0.98*
3	116.11 ± 0.1 (4)	*0.93*	116.42 ± 0.1 (3)	*0.71*	**116.50** ± 0.0 (1)	*1.00*	**116.50** ± 0.0 (2)	*1.00*
4	111.88 ± 0.8 (3)	*0.75*	**112.37** ± 0.7 (1)	*0.69*	111.59 ± 1.3 (4)	*1.00*	**112.19** ± 1.3 (2)	*0.98*
5	112.03 ± 0.1 (3)	*0.66*	111.86 ± 0.3 (4)	*0.63*	112.17 ± 0.0 (2)	*1.00*	**112.19** ± 0.0 (1)	*1.00*
6	113.86 ± 0.3 (3)	*0.88*	114.23 ± 0.2 (2)	*0.72*	**114.34** ± 0.1 (1)	*1.00*	113.02 ± 0.3 (4)	*0.99*
7	114.06 ± 0.1 (4)	*0.94*	114.32 ± 0.1 (3)	*0.66*	114.37 ± 0.0 (2)	*1.00*	**114.38** ± 0.0 (1)	*1.00*
8	110.70 ± 0.2 (4)	*0.79*	**111.24** ± 0.3 (1)	*0.67*	111.07 ± 0.1 (3)	*1.00*	111.14 ± 0.0 (2)	*1.00*
9	**111.70** ± 0.5 (1)	*0.68*	111.46 ± 0.1 (2)	*0.68*	110.19 ± 0.7 (3)	*1.00*	109.36 ± 2.9 (4)	*0.98*

set $p = 9$ and a computational budget of 10^7 MO-fevals. A population size of $N = 200$ was shown to work well for UHVEA [18], which we use here also for BezEA. We perform 30 runs, and a pair-wise Wilcoxon rank-sum test with $\alpha = 0.05$ is used to test whether differences with the best obtained result are statistically significant (up to 4 decimals). Ranks (in brackets) are computed based on the mean hypervolume values.

Results are given in Table 1. WFG1 is problematic, as none of the algorithms have an explicit mechanism to deal with its flat region. WFG2 has a disconnected Pareto front. MO-GOMEA* and UHVEA-gb both obtain solutions in multiple subsets, while BezEA obtains all solutions in a single connected subset, and spreads out well there. The linear front of WFG3 corresponds to the equidistant distribution of solutions along the Bézier curve, and BezEA outperforms the other methods there. Increasing q generally increases performance of BezEA, except for WFG6 and WFG9. Both these problems are non-separable, and require a larger population size than the currently used $N = 200$ to be properly solved. However, the guideline for non-separable problems results in a population size that is too large to be of practical relevance here. In terms of smoothness, BezEA with $q = 3$ is able to obtain a smoothness close to 1, while simultaneously obtaining the best HV_9 for 4/9 problems. MO-GOMEA* obtains a mean smoothness of 0.81 while UHVEA-gb obtains the worst mean smoothness (0.68). To illustrate the obtained smoothness a parallel coordinate plot for WFG7 is given in Fig. 5. This figure shows a clear pattern in decision variable values along the front (in the order **o**) for BezEA. This pattern is not obvious for the other two methods, while they achieve only a slightly lower hypervolume, and a lower smoothness.

6 Discussion and Outlook

In this work, we parameterized approximation sets as smooth Bézier curves in decision space, thereby explicitly enforcing a form of smoothness between

decision variables of neighboring solutions when the approximation front is traversed, aimed to improve its navigability. We used an UHV-based MO problem formulation that directly allows for the optimization of parameterized approximation sets. Solving this Bézier problem formulation with GOMEA (BezEA), was shown to be competitive to UHV-based optimization and domination-based MOEAs, while smoothness is guaranteed. We showed that approximation sets obtained with BezEA show a more clear pattern in terms of decision variables when traversing the approximation front on a set of benchmark problems, which suggests that this approach will lead to a more intuitive and smooth approximation set navigability for real-world optimization problems.

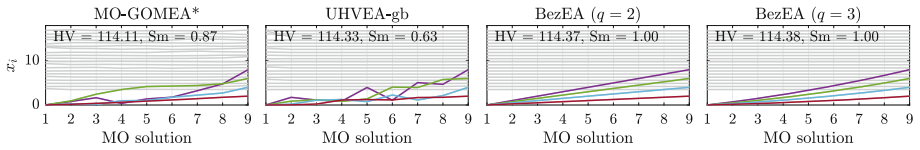

Fig. 5. Parallel coordinate plots shows of decision variables x_i for WFG7. In color the $k_{\text{WFG}} = 4$ position-type decision variables, in grey the remaining decision variables.

We chose to fix the solution set size p for BezEA during and after optimization, but since a parametric expression of the approximation set is available, it is straightforward to construct a large approximation set after optimization. This could be exploited to increase performance of BezEA, as it currently show computational overhead on the simple bi-sphere problem in terms of multi-objective function evaluations compared to UHVEA. In contrast to MOEAs, UHVEA and BezEA have the ability to converge to the Pareto set. When the problem is multimodal, UHVEA will spread its search over multiple modes. In that case, even an a posteriori fitting of a smooth curve through the obtained approximation set will result in low-quality solutions. BezEA on the other hand aims to obtain solutions in a single mode, thereby guaranteeing smoothness, even in a multimodal landscape. This form of regularization that is enforced upon approximation sets shows that BezEA can outperform MO-GOMEA* and UHVEA-gb on multiple problems in the WFG benchmark.

The smoothness measure introduced in this work is a measure for entire solution sets $\mathcal{S}_p$, and not for individual solutions $\mathbf{x}$. It can therefore not be added directly as an additional objective to the original multi-objective problem $\mathbf{f}(\mathbf{x})$. We chose in this work to introduce a parameterization of approximation sets that directly enforces smoothness. Alternatively, smoothness could also be added as a second objective to the UHV-based problem formulation. This then results in the pn-dimensional bi-objective optimization problem, given by $h(\mathcal{S}_p) = [\text{UHV}_{\mathbf{f}}(\mathcal{S}_p) \ ; \ \text{Sm}(\mathcal{S}_p)]$. This problem can then be solved with a domination-based MOEA, or even by again formulating it as a (much) higher-dimensional UHV-based single-objective problem. Whether this approach can be

efficient, even when grey-box properties such as partial evaluations are exploited, remains however future work.

The problems in this work were limited to problems involving two objectives. The presented results show that it is an interesting research avenue to extend this work to problems with more objectives. The Pareto front of non-degenerate problems with m objectives is an $m-1$-dimensional manifold. Instead of a one-dimensional Bézier curve, the Pareto set can then be modeled by an $(m-1)$-dimensional Bézier simplex [15]. For the navigation of higher-dimensional manifolds, a one-dimensional path through all obtained solutions could still be used. However, navigation would be performed might be problem specific and should be discussed with end-users. BezEA is applied to treatment planning of brachytherapy for prostate cancer, and results can be found in the supplementary of this work (online [17]).

Source code for the algorithms in this work is made available at [16].

Acknowledgments. This work was supported by the Dutch Research Council (NWO) through Gravitation Programme Networks 024.002.003. We furthermore acknowledge financial support of the Nijbakker-Morra Foundation for a high-performance computing system.

References

1. Auger, A., Hansen, N.: A restart CMA evolution strategy with increasing population size. In: Proceedings of the IEEE Congress on Evolutionary Computation - CEC 2005, pp. 1769–1776. IEEE Press (2005)
2. Beume, N., Naujoks, B., Emmerich, M.: SMS-EMOA: multiobjective selection based on dominated hypervolume. Eur. J. Oper. Res. **181**(3), 1653–1669 (2007)
3. Bhardwaj, P., Dasgupta, B., Deb, K.: Modelling the Pareto-optimal set using B-spline basis functions for continuous multi-objective optimization problems. Eng. Optim. **46**(7), 912–938 (2014)
4. Bosman, P.A.N., Grahl, J., Thierens, D.: Benchmarking parameter-free AMaLGaM on functions with and without noise. Evol. Comput. **21**(3), 445–469 (2013)
5. Bouter, A., Alderliesten, T., Witteveen, C., Bosman, P.A.N.: Exploiting linkage information in real-valued optimization with the real-valued gene-pool optimal mixing evolutionary algorithm. In: Proceedings of the Genetic and Evolutionary Computation Conference - GECCO 2017, pp. 705–712. ACM Press, New York (2017)
6. Bouter, A., Luong, N.H., Alderliesten, T., Witteveen, C., Bosman, P.A.N.: The multi-objective real-valued gene-pool optimal mixing evolutionary algorithm. In: Proceedings of the Genetic and Evolutionary Computation Conference - GECCO 2017, pp. 537–544. ACM Press, New York (2017)
7. Deb, K.: An efficient constraint handling method for genetic algorithms. Comput. Methods Appl. Mech. Eng. **186**(2), 311–338 (2000)
8. Deb, K.: Multi-objective Optimization. Wiley, Chichester (2001)
9. Fleischer, M.: The measure of Pareto optima applications to multi-objective metaheuristics. In: Fonseca, C.M., Fleming, P.J., Zitzler, E., Thiele, L., Deb, K. (eds.) EMO 2003. LNCS, vol. 2632, pp. 519–533. Springer, Heidelberg (2003). https://doi.org/10.1007/3-540-36970-8_37

10. Gallier, J.: Curves and Surfaces in Geometric Modeling: Theory and Algorithms. Morgan Kaufmann Publishers Inc., San Francisco (1999)
11. Guerreiro, A., Fonseca, C., Paquete, L.: Greedy hypervolume subset selection in low dimensions. Evol. Comput. **24**(3), 521–544 (2016)
12. Huband, S., Barone, L., While, L., Hingston, P.: A scalable multi-objective test problem toolkit. In: Coello Coello, C.A., Hernández Aguirre, A., Zitzler, E. (eds.) EMO 2005. LNCS, vol. 3410, pp. 280–295. Springer, Heidelberg (2005). https://doi.org/10.1007/978-3-540-31880-4_20
13. Knowles, J.: Local-search and hybrid evolutionary algorithms for Pareto optimization. Technical report, Ph.D. thesis, University of Reading (2002)
14. Knowles, J., Thiele, L., Zitzler, E.: A tutorial on the performance assessment of stochastic multiobjective optimization. Technical report, Computer Engineering and Networks Laboratory (TIK), ETH Zurich - TIK report 214 (2006)
15. Kobayashi, K., Hamada, N., Sannai, A., Tanaka, A., Bannai, K., Sugiyama, M.: Bezier simplex fitting: describing Pareto fronts of simplicial problems with small samples in multi-objective optimization. Preprint arXiv:1812.05222 (2018)
16. Maree, S.C.: Uncrowded-hypervolume multi-objective optimization C++ source code on Github (2019). https://github.com/scmaree/uncrowded-hypervolume
17. Maree, S.C., Alderliesten, T., Bosman, P.A.N.: Ensuring smoothly navigable approximation sets by Bézier curve parameterizations in evolutionary bi-objective optimization - applied to brachytherapy treatment planning for prostate cancer. Preprint arXiv:2006.06449 (2020)
18. Maree, S.C., Alderliesten, T., Bosman, P.A.N.: Uncrowded hypervolume-based multi-objective optimization with gene-pool optimal mixing. Preprint arXiv:2004.05068 (2020)
19. Mehta, V.K., Dasgupta, B.: Parametric approximation of the Pareto set in multi-objective optimization problems. J. Multi-Crit. Decis. Anal. **21**, 335–362 (2014)
20. Touré, C., Hansen, N., Auger, A., Brockhoff, D.: Uncrowded hypervolume improvement: COMO-CMA-ES and the sofomore framework. In: Proceedings of the Genetic and Evolutionary Computation Conference - GECCO 2019, pp. 638–646. ACM Press, New York (2019)
21. Wang, H., Deutz, A., Bäck, T., Emmerich, M.: Hypervolume indicator gradient ascent multi-objective optimization. In: Trautmann, H., et al. (eds.) EMO 2017. LNCS, vol. 10173, pp. 654–669. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-54157-0_44
22. Zitzler, E., Laumanns, M., Thiele, L.: SPEA2: improving the strength Pareto evolutionary algorithm for multiobjective optimization. In: Evolutionary Methods for Design, Optimisation and Control with Application to Industrial Problems - EUROGEN 2001, pp. 95–100. International Center for Numerical Methods in Engineering (CIMNE) (2001)
23. Zitzler, E., Thiele, L.: Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. IEEE Trans. Evol. Comput. **3**(4), 257–271 (1999)
24. Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C.M., Da Fonseca, V.G.: Performance assessment of multiobjective optimizers: an analysis and review. IEEE Trans. Evol. Comput. **7**(2), 117–132 (2003)