Bézier Simplex Fitting: Describing Pareto Fronts of Simplicial Problems with Small Samples in Multi-Objective Optimization

Ken Kobayashi,[1,2] **Naoki Hamada,**[1,2] **Akiyoshi Sannai,**[2,3] **Akinori Tanaka,**[2,3]
Kenichi Bannai,[3,2] **Masashi Sugiyama**[2,4]
[1]Artificial Intelligence Laboratory, Fujitsu Laboratories Ltd., Japan
[2]The Center for Advanced Intelligence Project, RIKEN, Japan
[3]Department of Mathematics, Faculty of Science and Technology, Keio University, Japan
[4]Department of Complexity Science and Engineering, Graduate School of Frontier Sciences, The University of Tokyo, Japan

Abstract

Multi-objective optimization problems require simultaneously optimizing two or more objective functions. Many studies have reported that the solution set of an M-objective optimization problem often forms an $(M-1)$-dimensional topological simplex (a curved line for $M = 2$, a curved triangle for $M = 3$, a curved tetrahedron for $M = 4$, etc.). Since the dimensionality of the solution set increases as the number of objectives grows, an exponentially large sample size is needed to cover the solution set. To reduce the required sample size, this paper proposes a Bézier simplex model and its fitting algorithm. These techniques can exploit the simplex structure of the solution set and decompose a high-dimensional surface fitting task into a sequence of low-dimensional ones. An approximation theorem of Bézier simplices is proven. Numerical experiments with synthetic and real-world optimization problems demonstrate that the proposed method achieves an accurate approximation of high-dimensional solution sets with small samples. In practice, such an approximation will be conducted in the post-optimization process and enable a better trade-off analysis.

Introduction

A multi-objective optimization problem is a problem that minimizes multiple objective functions $f_1, \dots, f_M : X \to \mathbb{R}$ over a common domain $X \subseteq \mathbb{R}^L$:

$$\begin{aligned} &\text{minimize } \boldsymbol{f}(\boldsymbol{x}) := (f_1(\boldsymbol{x}), \dots, f_M(\boldsymbol{x})) \\ &\text{subject to } \boldsymbol{x} \in X. \end{aligned}$$

Different functions usually have different minimizers, and one needs to consider a trade-off that two solutions $\boldsymbol{x}, \boldsymbol{y} \in X$ may satisfy $f_i(\boldsymbol{x}) < f_i(\boldsymbol{y})$ and $f_j(\boldsymbol{x}) > f_j(\boldsymbol{y})$. According to *Pareto ordering*, i.e.,

$$\begin{aligned} \boldsymbol{f}(\boldsymbol{x}) \prec \boldsymbol{f}(\boldsymbol{y}) \overset{\text{def}}{\iff} &\ f_m(\boldsymbol{x}) \le f_m(\boldsymbol{y}) \text{ for all } m = 1, \dots, M \\ &\text{and } f_m(\boldsymbol{x}) < f_m(\boldsymbol{y}) \text{ for some } m = 1, \dots, M, \end{aligned}$$

the goal of multi-objective optimization is to obtain the *Pareto set*

$$X^*(\boldsymbol{f}) := \{\, \boldsymbol{x} \in X \mid \boldsymbol{f}(\boldsymbol{y}) \not\prec \boldsymbol{f}(\boldsymbol{x}) \text{ for all } \boldsymbol{y} \in X \,\}$$

and the *Pareto front*

$$\boldsymbol{f}X^*(\boldsymbol{f}) := \left\{\, \boldsymbol{f}(\boldsymbol{x}) \in \mathbb{R}^M \;\middle|\; \boldsymbol{x} \in X^*(\boldsymbol{f}) \,\right\}$$

which describe the best-compromising solutions and their values of the conflicting objective functions, respectively.

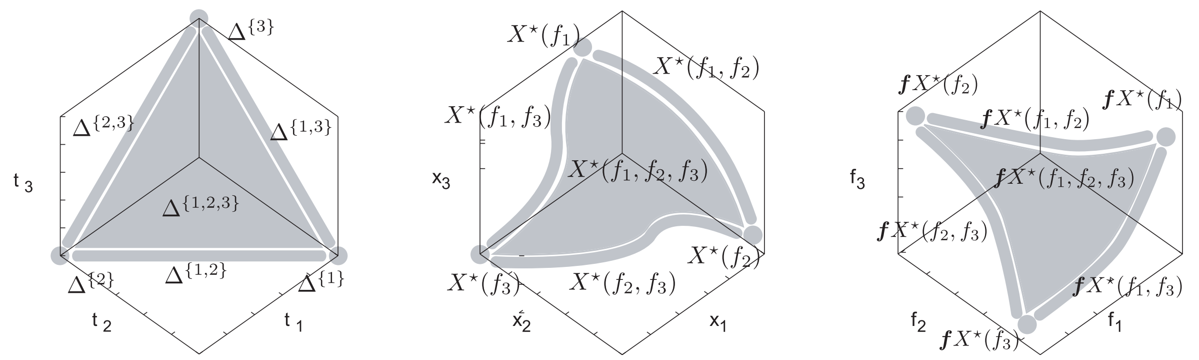
In industrial applications, obtaining the whole Pareto set/front rather than a single solution enables us to compare promising alternatives and to explore new innovative designs, whose concept is variously refered to as innovization (Deb and Srinivasan 2006), multi-objective design exploration (Obayashi, Jeong, and Chiba 2005) and design informatics (Chiba, Makino, and Takatoya 2009). Quite a few real-world problems involve simulations and/or experiments to evaluate solutions (Chand and Wagner 2015) and lack the mathematical expression of their objective functions and derivatives. Multi-objective evolutionary algorithms are a tool to solve such problems where the Pareto set/front is approximated by a population, i.e., a finite set of sample points (Coello, Lamont, and Van Veldhuisen 2007).

While the available sample size is limited due to expensive simulations and experiments, it is well-known that the dimensionality of the Pareto set/front increases as the number of objectives grows. Describing high-dimensional Pareto sets/fronts with small samples is one of the key challenges in many-objective optimization today (Li et al. 2015).

A considerable number of real-world applications share an interesting structure: their Pareto sets and/or Pareto fronts are often homeomorphic to an $(M-1)$-dimensional simplex. See for example (Rodríguez-Chía and Puerto 2002; Vrugt et al. 2003; Shoval et al. 2012; Mastroddi and Gemma 2013). This observation has been theoretically backed up in some cases (Kuhn 1967; Smale 1973; Shoval et al. 2012). A recent study (Lovison and Pecci 2014) pointed out that for all $m \le M$, each $(m-1)$-dimensional face of such a simplex is the Pareto set of a subproblem optimizing m objective functions of the original problem.

By exploiting this simplex structure, we considers the problem of fitting a hyper-surface to the Pareto front. In statistics, machine learning and related fields, regression problems have been considered in Euclidean space without boundary or at most with coordinate-wise upper/lower boundaries (Gelman and Hill 2007; Gelman et al. 2013). Hyper-surface models developed in such spaces are not suitable for Pareto fronts as a simplex has a non-axis-parallel boundary whose skeleton structure is described by its faces.

This paper proposes a new model and its fitting algorithm for approximating Pareto fronts with the simplex structure.



(a) Simplex Δ^J (b) Pareto set $X^*(\boldsymbol{f}_J)$ (c) Pareto front $\boldsymbol{f}X^*(\boldsymbol{f}_J)$

Figure 1: Skeletons for non-empty $J \subseteq \{ 1, 2, 3 \}$ and simplicial $\boldsymbol{f} = (f_1, f_2, f_3)$.

Our contibution can be summarized as follows:

1. We define a new class of multi-objective optimization problems called the simplicial problem in which the Pareto set/front have the simplex structure discussed above. We propose a Bézier simplex model, which is a generalization of a Bézier curve (Farin 2002).
2. We prove that Bézier simplices can approximate the Pareto set/front (as well as the objective map between them) of any simplicial problem with arbitrary accuracy.
3. We propose a Bézier simplex fitting algorithm. Exploiting the simplex structure of the Pareto set/front of a simplicial problem, this algorithm decomposes a Bézier simplex into low-dimensional simplices and fits each of them, inductively. This approach allows us to reduce the number of parameters to be estimated at a time.
4. We evaluate the approximation accuracy of the proposed method with synthetic and real-world optimization problems; compared to a conventional response surface model for Pareto fronts, our method exhibits a better boundary approximation while keeping the almost same quality of interior approximation. As a result, a five-objective Pareto front is described with only tens of sample points.

Preliminaries

Let us introduce notations for defining simplicial problems and review an existing method of Bézier curve fitting.

Simplicial Problem

A multi-objective optimization problem is denoted by its objective map $\boldsymbol{f} = (f_1, \dots, f_M) : X \to \mathbb{R}^M$. Let $I := \{ 1, \dots, M \}$ be the index set of objective functions and

$$\Delta^{M-1} := \left\{ (t_1, \dots, t_M) \in \mathbb{R}^M \;\middle|\; 0 \le t_m, \sum_{m \in M} t_m = 1 \right\}$$

be the *standard simplex* in $\mathbb{R}^M$. For each non-empty subset $J \subseteq I$, we call

$$\Delta^J := \{ (t_1, \dots, t_M) \in \Delta^{M-1} \mid t_m = 0 \ (m \notin J) \}$$

the *J-face* of Δ^{M-1} (see Figure 1a) and

$$\boldsymbol{f}_J := (f_i)_{i \in J} : X \to \mathbb{R}^{|J|}$$

the *J-subproblem* of $\boldsymbol{f}$. For each $0 \le m \le M-1$, we call

$$\Delta^{(m)} := \bigcup_{J \subseteq I,\ |J| = m} \Delta^J$$

the *m-skeleton* of Δ^{M-1}.

The problem class we are interested in is as follows:

Definition 1 (Simplicial problem) *A problem $\boldsymbol{f} : X \to \mathbb{R}^M$ is* simplicial *if there exists a map $\boldsymbol{\phi} : \Delta^{M-1} \to X$ such that for each non-empty subset $J \subseteq I$, its restriction $\boldsymbol{\phi}|_{\Delta^J} : \Delta^J \to X$ gives homeomorphisms*

$$\boldsymbol{\phi}|_{\Delta^J} : \Delta^J \to X^*(\boldsymbol{f}_J),$$
$$\boldsymbol{f} \circ \boldsymbol{\phi}|_{\Delta^J} : \Delta^J \to \boldsymbol{f}X^*(\boldsymbol{f}_J).$$

We call such $\boldsymbol{\phi}$ and $\boldsymbol{f} \circ \boldsymbol{\phi}$ a triangulation *of the Pareto set $X^*(\boldsymbol{f})$ and the Pareto front $\boldsymbol{f}X^*(\boldsymbol{f})$, respectively. For each non-empty subset $J \subseteq I$, we call $X^*(\boldsymbol{f}_J)$ the J*-face *of $X^*(\boldsymbol{f})$ and $\boldsymbol{f}X^*(\boldsymbol{f}_J)$ the J*-face *of $\boldsymbol{f}X^*(\boldsymbol{f})$. For each $0 \le m \le M-1$, we call*

$$X^{*(m)} := \bigcup_{J \subseteq I,\ |J| = m} X^*(\boldsymbol{f}_J),$$
$$\boldsymbol{f}X^{*(m)} := \bigcup_{J \subseteq I,\ |J| = m} \boldsymbol{f}X^*(\boldsymbol{f}_J)$$

the m-skeleton *of $X^*(\boldsymbol{f})$ and $\boldsymbol{f}X^*(\boldsymbol{f})$, respectively.*

By definition, any subproblem of a simplicial problem is again simplicial. The homeomorphism $\boldsymbol{\phi}|_{\Delta^J}$ ensures that the Pareto sets forms a curved simplex as shown in Figure 1b. The homeomorphism $\boldsymbol{f} \circ \boldsymbol{\phi}|_{\Delta^J}$ asserts that $\boldsymbol{f}|_{X^*(\boldsymbol{f})} : X^*(\boldsymbol{f}) \to \mathbb{R}^M$ is a C^0-embedding. This means that the Pareto front is also a curved simplex (with a possibly different shape) as shown in Figure 1c. Therefore, the Pareto set/front of an M-objective simplicial problem can be treated as a curved $(M-1)$-simplex. We can find its J-face by solving the J-subproblem.

The above structure appears in a broad range of applications. In operations research, the Pareto set of the facility location problem under the L^2-norm is shown to be the convex hull of single-objective optima (Kuhn 1967). When the optima are in general position, the Pareto set becomes a simplex. Similar observations are also reported under other norms (Rodríguez-Chía and Puerto 2002). In economics, the Pareto set of the pure exchange economy with M players is known to be homeomorphic to an $(M-1)$-dimensional simplex (Smale 1973). In hydrology, the two-objective Pareto set of a hydrologic cycle model calibration is observed to be a curve. Its end points are single-objective optima, and the end points correspond to end points of the Pareto front curve (Vrugt et al. 2003). In addition, a recent study pointed out that the Pareto set of an M-objective convex optimization problem is diffeomorphic to an $(M-1)$-dimensional simplex and that its $(m-1)$-dimensional faces are the Pareto sets of m-objective subproblems for all $m \leq M$ (Lovison and Pecci 2014).

Bézier Curve Fitting

Since the Pareto front of any two-objective simplicial problem is a curve with two end points in $\mathbb{R}^3$, the Bézier curve would be a suitable model for describing it.

In $\mathbb{R}^M$, the Bézier curve of degree D is a parametric curve, i.e., a map $\boldsymbol{b} : [0,1] \to \mathbb{R}^M$ determined by $D+1$ *control points* $\boldsymbol{p}_0, \dots, \boldsymbol{p}_D \in \mathbb{R}^M$ (Farin 2002):

$$\boldsymbol{b}(t) := \sum_{d=0}^{D} \binom{D}{d} t^d (1-t)^{(D-d)} \boldsymbol{p}_d \quad (0 \leq t \leq 1), \tag{1}$$

where $\binom{D}{d}$ represents the binomial coefficient. The parameter t moves from $t=0$ to $t=1$, giving a curve $\boldsymbol{b}(t)$ with two end points $\boldsymbol{b}(0) = \boldsymbol{p}_0$ and $\boldsymbol{b}(1) = \boldsymbol{p}_D$.

Given sample points $\boldsymbol{x}_1, \dots, \boldsymbol{x}_N \in \mathbb{R}^M$, a Bézier curve can be fitted by solving the following problem (Borges and Pastva 2002):

$$\begin{aligned} &\underset{t_n,\, \boldsymbol{p}_d}{\text{minimize}} \sum_{n=1}^{N} \|\boldsymbol{b}(t_n) - \boldsymbol{x}_n\|^2 \\ &\text{subject to } \ 0 \leq t_n \leq 1 \quad (n = 1, \dots, N) \end{aligned} \tag{2}$$

where t_n $(n = 1, \dots, N)$ and $\boldsymbol{p}_d$ $(d = 0, \dots, D)$ are variables to be optimized. Notice that t_n $(n = 1, \dots, N)$ are introduced to calculate residuals for each sample point. The error function (2) to be minimized represents the sum of squared residuals for sample points.

If one fix all control points $\boldsymbol{p}_d$ $(d = 0, \dots, D)$, the Bézier curve $\boldsymbol{b}(t)$ is determined. The error function (2) can be now separately optimized by minimizing $\|\boldsymbol{b}(t_n) - \boldsymbol{x}_n\|^2$ with respect to each t_n. The solution t_n is the foot of a perpendicular line from a sample point $\boldsymbol{x}_n$ to the Bézier curve $\boldsymbol{b}([0,1])$, which satisfies

$$\left\langle \left.\frac{\partial}{\partial t}\right|_{t=t_n} \boldsymbol{b}(t),\ \boldsymbol{b}(t_n) - \boldsymbol{x}_n \right\rangle = 0. \tag{3}$$

Since (3) is a nonlinear equation, Newton's method is used to find the solution t_n.

Algorithm 1 Bézier curve fitting (Borges and Pastva 2002)

```
(Initialize) Set $i \leftarrow 1$ and initial control points $\boldsymbol{p}_d^{(i)}$ $(d = 0, \dots, D)$.
while not converged do
    (Update parameters) Fix control points $\boldsymbol{p}_d^{(i)}$ $(d = 0, \dots, D)$ and solve (3) for each $n = 1, \dots, N$ using Newton's method. Then set the solutions as $t_n^{(i+1)}$.
    (Update control points) Solve (2) with respect to the control points and set the solutions as $\boldsymbol{p}_d^{(i+1)}$ $(d = 0, \dots, D)$.
    $i \leftarrow i + 1$.
end while
return $\boldsymbol{p}_d^{(i)}$ $(d = 0, \dots, D)$
```

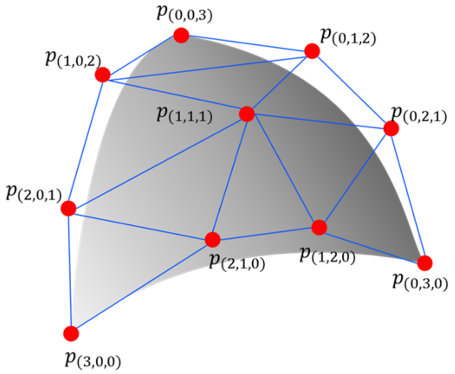


Figure 2: A Bézier simplex for $M = 3$, $D = 3$.

If one fix all parameters t_n $(n = 1, \dots, N)$, the Bézier curve $\boldsymbol{b}(t_n)$ becomes a linear function with respect to the control points $\boldsymbol{p}_0, \dots, \boldsymbol{p}_D$. The error function (2) can be now optimized by solving linear equations with respect to all $\boldsymbol{p}_d$.

Algorithm 1 shows Borges and Pastva's method, which alternately adjusts parameters and control points. This algorithm is also used to adjust some of the control points with remaining points fixed (Shao and Zhou 1996).

Bézier Simplex Fitting

To describe the Pareto front of an arbitrary-objective simplicial problem, however, Bézier curve fitting is not enough, and we need to generalize it to the Bézier simplex. We propose a method of fitting a Bézier simplex to the Pareto front of a simplicial problem.

Bézier Simplex

Let $\mathbb{N}$ be the set of nonnegative integers and

$$\mathbb{N}_D^M := \left\{ (d_1, \dots, d_M) \in \mathbb{N}^M \;\middle|\; \sum_{m=1}^{M} d_m = D \right\}.$$

For $\boldsymbol{t} := (t_1, \dots, t_M) \in \mathbb{R}^M$ and $\boldsymbol{d} := (d_1, \dots, d_M) \in \mathbb{N}^M$, we denote by $\boldsymbol{t}^{\boldsymbol{d}}$ a monomial $t_1^{d_1} t_2^{d_2} \cdots t_M^{d_M}$. As shown in Figure 2, the Bézier simplex of degree D in $\mathbb{R}^M$ is a map

$\Delta^{M-1} \to \mathbb{R}^M$ determined by control points $\boldsymbol{p_d} \in \mathbb{R}^M$ ($\boldsymbol{d} \in \mathbb{N}_D^M$):

$$\boldsymbol{b}(\boldsymbol{t}) := \sum_{\boldsymbol{d}\in\mathbb{N}_D^M} \binom{D}{\boldsymbol{d}} \boldsymbol{t}^{\boldsymbol{d}} \boldsymbol{p_d}, \tag{4}$$

where $\binom{D}{\boldsymbol{d}}$ represents a polynomial coefficient

$$\binom{D}{\boldsymbol{d}} := \frac{D!}{d_1!d_2!\cdots d_M!}.$$

Approximation Theorem

Is the Bézier simplex a suitable model for describing the Pareto front of a simplicial problem? Let us check that for any simplicial problem, Bézier simplices can approximate the Pareto front (as well as the Pareto set and the objective map between them) with arbitrary accuracy.

We begin with a more general proposition that any continuous map on a simplex can be uniformly approximated by some Bézier simplex as a map:

Theorem 1 *Let $\boldsymbol{\phi} : \Delta^{M-1} \to \mathbb{R}^K$ be a continuous map. There exists an infinite sequence of Bézier simplices $\boldsymbol{b}^{(i)} : \Delta^{M-1} \to \mathbb{R}^K$ such that*

$$\lim_{i\to\infty} \sup_{\boldsymbol{t}\in\Delta^{M-1}} |\boldsymbol{\phi}(\boldsymbol{t}) - \boldsymbol{b}^{(i)}(\boldsymbol{t})| = 0.$$

The proof of Theorem 1 is shown in Appendix A.[1]

Recall (Definition 1) that a simplicial problem $\boldsymbol{f} : X \to \mathbb{R}^M$ admits a triangulation of the Pareto set

$$\boldsymbol{\phi} : \Delta^{M-1} \to X^*(\boldsymbol{f}) \subset \mathbb{R}^L,$$

which induces a triangulation of the Pareto front

$$\boldsymbol{f} \circ \boldsymbol{\phi} : \Delta^{M-1} \to \boldsymbol{f}X^*(\boldsymbol{f}) \subset \mathbb{R}^M.$$

Addtionally, the trianglation $\boldsymbol{\phi} : \Delta^{M-1} \to X^*(\boldsymbol{f})$ also induces a triangulation of a restricted objective map $\boldsymbol{f} : X^*(\boldsymbol{f}) \to \boldsymbol{f}X^*(\boldsymbol{f})$. That is, its graph $G^*(\boldsymbol{f}) = \left\{ (\boldsymbol{x}, \boldsymbol{f}(\boldsymbol{x})) \in \mathbb{R}^L \times \mathbb{R}^M \;\middle|\; \boldsymbol{x} \in X^*(\boldsymbol{f}) \right\}$ is triangulated by the induced map

$$(\boldsymbol{\phi}, \boldsymbol{f} \circ \boldsymbol{\phi}) : \Delta^{M-1} \to G^*(\boldsymbol{f}) \subset \mathbb{R}^{L+M}$$

because $\boldsymbol{f} : X^*(\boldsymbol{f}) \to \boldsymbol{f}X^*(\boldsymbol{f})$ is continuous by definition and any continuous map induces a homeomorphism from the domain to the graph. These three triangulations fulfill the qualifications of the map $\boldsymbol{\phi}$ in Theorem 1. We thus get the desired result for approximation as a set.

Corollary 2 *Let X^* be the Pareto set, the Pareto front or the graph of the objective map restricted to the Pareto set of a simplicial problem. There exists an infinite sequence of Bézier simplices $\boldsymbol{b}^{(i)} : \Delta^{M-1} \to \mathbb{R}^K$ such that*

$$\lim_{i\to\infty} d_{\mathrm{H}}(X^*, B^{(i)}) = 0$$

where d_{H} is the Hausdorff distance and $B^{(i)}$ are images of Bézier simplices: $B^{(i)} := \boldsymbol{b}^{(i)}(\Delta^{M-1})$.

Algorithm 2 Bézier simplex fitting (all-at-once)

```
(Initialize) Set $i \leftarrow 1$ and initial control points $\boldsymbol{p}_{\boldsymbol{d}}^{(i)}$ ($\boldsymbol{d} \in \mathbb{N}_D^M$).
while not converged do
    (Update parameters) Fix control points $\boldsymbol{p}_{\boldsymbol{d}}^{(i)}$ ($\boldsymbol{d} \in \mathbb{N}_D^M$) and solve (6) for each $n = 1, \dots, N$ using Newton's method. Then set the solutions as $\boldsymbol{t}_n^{(i+1)}$.
    (Update control points) Solve (5) with respect to the control points and set the solutions $\boldsymbol{p}_{\boldsymbol{d}}^{(i+1)}$ ($\boldsymbol{d} \in \mathbb{N}_D^M$).
    $i \leftarrow i + 1$.
end while
return $\boldsymbol{p}_{\boldsymbol{d}}^{(i)}$ ($\boldsymbol{d} \in \mathbb{N}_D^M$)
```

All-at-Once Fitting

Let us consider algorithms to realize such a sequence of Bézier simplices. First, we develop a straightforward generalization of the Bézier curve fitting method. We call this method the *all-at-once fitting*.

Given sample points $\boldsymbol{x}_1, \dots, \boldsymbol{x}_N \in \mathbb{R}^M$, a Bézier simplex can be fitted by solving the following problem, which is a multi-dimensional analogue of the problem (2):

$$\begin{aligned} &\underset{\boldsymbol{t}_n,\ \boldsymbol{p_d}}{\text{minimize}} \sum_{n=1}^{N} \|\boldsymbol{b}(\boldsymbol{t}_n) - \boldsymbol{x}_n\|^2 \\ &\text{subject to } \boldsymbol{t}_n \in \Delta^{M-1} \quad (n = 1, \dots, N) \end{aligned} \tag{5}$$

where $\boldsymbol{t}_n = (t_{n1}, \dots, t_{nM})$ ($n = 1, \dots, N$) and $\boldsymbol{p_d}$ ($\boldsymbol{d} \in \mathbb{N}_D^M$) are variables to be optimized.

As is the case of Bézier curve fitting, if one fix all control points $\boldsymbol{p_d}$ ($\boldsymbol{d} \in \mathbb{N}_D^M$), the Bézier simplex $\boldsymbol{b}(\boldsymbol{t})$ is determined. The error function (5) can be now separately optimized by minimizing $\|\boldsymbol{b}(\boldsymbol{t}_n) - \boldsymbol{x}_n\|^2$ with respect to each $\boldsymbol{t}_n$. The solution $\boldsymbol{t}_n$ is the foot of a perpendicular line from a sample point $\boldsymbol{x}_n$ to the Bézier simplex $\boldsymbol{b}(\Delta^{M-1})$, which satisfies

$$\left\langle \left.\frac{\partial}{\partial t_m}\right|_{\boldsymbol{t}=\boldsymbol{t}_n} \boldsymbol{b}(\boldsymbol{t}),\ \boldsymbol{b}(\boldsymbol{t}_n) - \boldsymbol{x}_n \right\rangle = 0 \quad (m = 1, \dots, M). \tag{6}$$

Since (6) is a system of nonlinear equations, Newton's method is used to find the solution $\boldsymbol{t}_n$.

If one fix all parameters $\boldsymbol{t}_n$ ($n = 1, \dots, N$), the Bézier simplex $\boldsymbol{b}(\boldsymbol{t}_n)$ is a linear function with respect to the control points $\boldsymbol{p_d}$ ($\boldsymbol{d} \in \mathbb{N}_D^M$). The error function (5) can be now optimized by solving linear equations with respect to all $\boldsymbol{p_d}$ ($\boldsymbol{d} \in \mathbb{N}_D^M$).

As well as Borges and Pastva's method, the all-at-once fitting alternately adjusts parameters and control points. We describe the all-at-once fitting in Algorithm 2. This algorithm is also used to adjust some of the control points with remaining control points fixed.

While a Bézier simplex has enough flexibility as described in Theorem 1, the required sample size for the all-at-once fitting grows quickly with respect to M. Let us consider the case of fitting a Bézier simplex of degree D to the Pareto front of an M-objective problem. In this case, the number of

[1] Appendix is available from https://github.com/rafcc/stratification-learning.

control points to be estimated is

$$\left|\mathbb{N}_D^M\right| = \binom{D+M-1}{D} = \frac{(D+M-1)!}{D!(M-1)!} = O(M^D).$$

Practically, the degree of a Bézier simplex to be fitted is often set as $D = 3$; nevertheless the number of control points to be estimated at a time becomes unreasonable for large M.

Inductive Skeleton Fitting

To reduce the required sample size, we consider decomposing a Bézier simplex into subsimlices and fitting each subsimplex one by one from low dimension to high dimension. This approach allows us to reduce the number of control points to be estimated at a time. We call this method the *inductive skeleton fitting*.

Let $I := \{ 1, \dots, M \}$ and for each non-empty subset $J \subseteq I$, we define

$$\mathbb{N}_D^J := \left\{ (d_1, \dots, d_M) \in \mathbb{N}_D^M \;\middle|\; d_m = 0 \ (m \notin J) \right\}.$$

The J-face of an $(M-1)$-Bézier simplex of degree D is a map $\Delta^J \to \mathbb{R}^M$ determined by control points $\boldsymbol{p_d} \in \mathbb{R}^M$ for all $\boldsymbol{d} \in \mathbb{N}_D^J$:

$$\boldsymbol{b}^J(\boldsymbol{t}) := \sum_{\boldsymbol{d} \in \mathbb{N}_D^J} \binom{D}{\boldsymbol{d}} \boldsymbol{t}^{\boldsymbol{d}} \boldsymbol{p_d}. \qquad (7)$$

For arbitrary parameters $\boldsymbol{t}$ satisfying

$$\boldsymbol{t} \in \Delta^J \subseteq \Delta^I,$$

all entries that are not in J are 0. It then holds that

$$\boldsymbol{b}^I(\boldsymbol{t}) = \sum_{\boldsymbol{d} \in \mathbb{N}_D^I} \binom{D}{\boldsymbol{d}} \boldsymbol{t}^{\boldsymbol{d}} \boldsymbol{p_d} = \sum_{\boldsymbol{d} \in \mathbb{N}_D^J} \binom{D}{\boldsymbol{d}} \boldsymbol{t}^{\boldsymbol{d}} \boldsymbol{p_d} = \boldsymbol{b}^J(\boldsymbol{t}).$$

This means that, for each non-empty subset $J \subset I$, the J-face of the $(M-1)$-Bézier simplex of degree D is the $(|J|-1)$-Bézier simplex of the same degree and it is determined by the control points $\boldsymbol{p_d}$ of the $(M-1)$-Bézier simplex satisfying $\boldsymbol{d} \in \mathbb{N}_D^J$. To exploiting this structure, the inductive skeleton fitting decomposes a Bézier simplex into subsimplices and fits a subsimplex $\boldsymbol{b}^J(\boldsymbol{t})$ to X_J for each non-empty subset $J \subseteq I$ in ascending order of cardinality of J.

When a problem $\boldsymbol{f} = (f_1, \dots, f_M) : X \to \mathbb{R}^M$ is simplicial, we can provide a set of subsamples for running the inductive skeleton fitting. Remember that the Pareto front $\boldsymbol{f}X^*(\boldsymbol{f})$ has the same skeleton as the $(M-1)$-simplex: $\boldsymbol{f}X^*(\boldsymbol{f}_J) \subseteq \boldsymbol{f}X^*(\boldsymbol{f}_I)$ for all $\emptyset \neq J \subseteq I$. Thus given a sample X of $\boldsymbol{f}X^*(\boldsymbol{f})$, we can decompose it into subsamples

$$X_J := \{ \boldsymbol{x} \in X \mid \boldsymbol{f}_J(y) \not\prec \boldsymbol{f}_J(x) \text{ for all } \boldsymbol{y} \in X \},$$

each of which represents a sample of the J-face $\boldsymbol{f}X^*(\boldsymbol{f}_J)$ $(\emptyset \neq J \subseteq I)$. Algorithm 3 summarizes the inductive skeleton fitting.

Unlike the all-at-once fitting, the inductive skeleton fitting allows us to reduce the number of control points to be estimated at a time. Let us consider the case of fitting a subsimplex $\boldsymbol{b}^J(\boldsymbol{t})$ with $|J| = m$. As we described before, the subsimplex $\boldsymbol{b}^J(\boldsymbol{t})$ has $|\mathbb{N}_D^m| = \binom{D+m-1}{D} = \frac{(D+m-1)!}{D!(m-1)!}$ control points. In practice, it is sufficient to set D as a small value compared to M. In such a case, the inductive skeleton fitting estimates at most D-objective solutions, then we only have to adjust at most $\max_{m=1,\dots,D} \binom{D+m-1}{D}$ control points for each step. Notice that this number does not depend on M but on D. Therefore, in case of the inductive skeleton fitting, the number of control points to be estimated at a time is much smaller than $\binom{D+M-1}{D}$ for large M.

Algorithm 3 Bézier simplex fitting (inductive skeleton)

```
for $m = 1, \dots, \min\{ D, M \}$ do
    for $J \subseteq I$ such that $|J| = m$ do
        Fix control points $\boldsymbol{p_d}$ $(\boldsymbol{d} \in \bigcup_{\emptyset \neq K \subset J} \mathbb{N}_D^K)$ as estimated in previous steps.
        Adjust remaining control points $\boldsymbol{p_d}$ $(\boldsymbol{d} \in \mathbb{N}_D^J \setminus (\bigcup_{\emptyset \neq K \subset J} \mathbb{N}_D^K))$ by Algorithm 2 with sample $X_J$.
    end for
end for
return $\boldsymbol{p_d}$ $(\boldsymbol{d} \in \mathbb{N}_D^M)$
```

Numerical Experiments

The small sample behavior of the proposed method is examined using Pareto front samples of varying size.[2]

Data Sets

To investigate the effect of the Pareto front shape and dimensionality, we employed six synthetic problems with known Pareto fronts, all of which are simplicial problems. Schaffer, ConstrEx and Osyczka2 are two-objective problems. Their Pareto fronts are a curved line that can be triangulated into two vertices and one edge. 3-MED and Viennet2 are three-objective problems. Their Pareto fronts are a curved triangle that can be triangulated into three vertices, three edges and one face. 5-MED is a five-objective problem. Its Pareto front is a curved pentachoron that can be triangulated into five vertices, ten edges, ten faces, five three-dimensional faces and one four-dimensional face. We generated Pareto front samples of 3-MED and 5-MED by AWA(objective) using default hyper-parameters (Hamada et al. 2010). Pareto front samples of the other problems were taken from jMetal 5.2 (Nebro, Durillo, and Vergne 2015).

To assess the practicality of the proposed method, we also used one real-world problem called S3TD. This is a four-objective problem[3] of designing a silent super-sonic aircraft. The Pareto front has not been exactly known, and we only have an inaccurate sample of 58 points obtained in the prior study (Chiba, Makino, and Takatoya 2009). Its simpliciality is also unknown. If it is assumed to be simplicial, then

[2] The source code is available from https://github.com/rafcc/stratification-learning.

[3] The original problem is five-objective, but we dropped one objective because there are two objectives that are highly correlated.

the Pareto front is a curved tetrahedron that can be triangulated into four vertices, six edges, four faces, one three-dimensional face. The problem definitions of the synthetic and real-world problems are described in Appendix B.

For each problem, the Pareto front sample is split into a training set and a validation set. The training set is further decomposed into subsamples for the Pareto fronts of subproblems: each m-objective subproblem has a subsample consisting of N_m points. Experiments were conducted on all combinations of the following subsample sizes:[4]

$$\begin{aligned} N_1 &= 1, \\ N_2 &= 2, \dots, 10, \\ N_3 &= 1, \dots, 10 \quad (N_4 = N_5 = 0). \end{aligned}$$

An M-objective problem has M single-objective problems, $M(M-1)/2$ two-objective problems and $M(M-1)(M-2)/6$ three-objective problems. The total sample size of the training set is

$$N = MN_1 + M(M-1)N_2/2 + M(M-1)(M-2)N_3/6.$$

The details of making Pareto front subsamples are described in Appendix C.

Methods

We compared the following surface-fitting methods:

- the inductive skeleton fitting (Algorithm 3);
- the all-at-once fitting (Algorithm 2);
- the response surface method (Goel et al. 2007).

The inductive skeleton fitting used each training subsample for each subproblem. The all-at-once fitting and the response surface method used all training subsamples as a whole. We set the initial control points $\boldsymbol{p}^0_{\boldsymbol{d}}$ $(\boldsymbol{d} \in \mathbb{N}^J_D,\ J \subset I,\ |J| = 1)$, which are the vertices of the Bézier simplex, to be the single-objective optima, and the rest of control points were set to be the simplex grid spanned by them. Newton's method used in the inductive skeleton fitting and the all-at-once fitting employed the first and second analytical derivatives, and was terminated when the number of iterations reached 100 or when the following condition is satisfied:

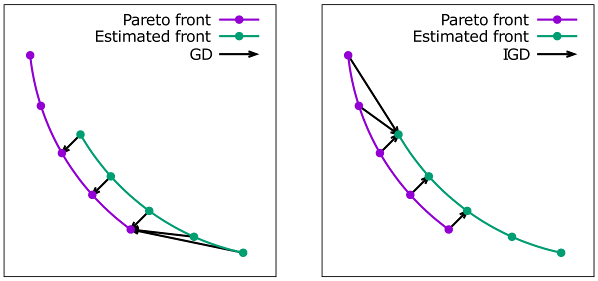
$$\sqrt{\sum_{m=1}^{M} \left\langle \frac{\partial}{\partial t_m}\bigg|_{\boldsymbol{t}=\boldsymbol{t}_n} \boldsymbol{b}(\boldsymbol{t}),\ \boldsymbol{b}(\boldsymbol{t}_n) - \boldsymbol{x}_n \right\rangle^2} \le 10^{-5}.$$

The fitting algorithm was terminated when the number of iterations reached 100 or when the following condition is satisfied:

$$\sqrt{\mathrm{SSR}_{i+1}}/N - \sqrt{\mathrm{SSR}_i}/N \le 10^{-5},$$

where SSR_i is the value of the loss function (5) at the i-th iteration.

According to a prior study (Goel et al. 2007), the response surface method treated multi-dimensional objective values as a function from the first $M-1$ objective values to the last objective value by using multi-polynomial regression with cubic polynomials. We removed cross terms of degree three so that the number of explanatory variables did not exceed the sample size.



(a) GD assesses false positive. (b) IGD assesses false negative.

Figure 3: GD and IGD.

Performance Measures

To evaluate how accurately an estimated hyper-surface approximates the Pareto front, we used the generational distance (GD) (Veldhuizen 1999) and the inverted generational distance (IGD) (Zitzler et al. 2003):

$$\mathrm{GD}(X, Y) := \frac{1}{|X|} \sum_{\boldsymbol{x} \in X} \min_{\boldsymbol{y} \in Y} \|\boldsymbol{x} - \boldsymbol{y}\|,$$

$$\mathrm{IGD}(X, Y) := \frac{1}{|Y|} \sum_{\boldsymbol{y} \in Y} \min_{\boldsymbol{x} \in X} \|\boldsymbol{x} - \boldsymbol{y}\|$$

where X is a finite set of points sampled from an estimated hyper-surface and Y is a validation set.

Figure 3 depicts what GD and IGD assess. These values can be viewed as the avarage length of arrows in each plot. Figure 3a implies that GD becomes high when the estimated front has a false positive area which is far from the Pareto front. Conversely, Figure 3b tells that IGD becomes high when the Pareto front has a false negative area which is not covered by the estimated front. Thus, we can say that the estimated hyper-surface is close to the Pareto front if and only if both GD and IGD are small.

For the all-at-once fitting and the inductive skeleton fitting, we sampled the estimated Bézier simplex as follows:

$$X := \left\{ \boldsymbol{b}(\boldsymbol{t}) \ \middle|\ \boldsymbol{t} \in \Delta^{M-1},\ t_m \in \left\{ 0, \frac{1}{20}, \frac{2}{20}, \dots, 1 \right\} \right\}.$$

For the response surface method, we sampled the estimated response surface as follows:

$$\begin{aligned} X := \{&(x_1, \dots, x_{M-1}, r(x_1, \dots, x_{M-1})) \\ &| x_1, \dots, x_{M-1} \in \{0, \frac{1}{20}, \frac{2}{20}, \dots, 1\}\}, \end{aligned}$$

where $r(x_1, \dots, x_{M-1})$ is an estimated polynomial function. We repeated experiments 20 times[5] with different training sets and computed the average and the standard deviations of their GDs and IGDs.

[4] We fix $N_4 = N_5 = 0$ as all methods used in our experiments do not require these subsamples to fit the Pareto front.

[5] We set the number of trials on the basis of power analysis. According to our preliminary experiments, the effect size of GD/IGD values was approximately $d = 0.6$. In this case, the sample size $n = 20$ is required to assume the power > 0.8 with significance level $p = 0.05$.

Table 1: GD and IGD (avg. ± s.d. over 20 trials) with subsample size $(1, 3)$ for Schaffer, ConstrEx and Osyczka2; $(1, 2, 1)$ for 3-MED, Viennet2 and 5-MED; $(1, 5, 5)$ for S3TD. The best scores are shown in bold. ($^{**} : p < 0.05, ^{*} : p < 0.1$)

Problem	**Inductive skeleton**			**All-at-once**			**Response surface**	
	Iteration	GD	IGD	Iteration	GD	IGD	GD	IGD
Schaffer	3.00 ± 0.00	5.21e-10 ± 1.16e-09	**2.49e-02 ± 1.68e-12**	3.00 ± 0.00	**5.21e-10 ± 1.16e-09**	**2.49e-02 ± 1.68e-12**	1.68e-01 ± 1.74e-01	1.13e-01 ± 3.50e-02
ConstrEx	3.00 ± 0.00	**4.94e-02 ± 4.73e-02**	**5.44e-02 ± 2.39e-02**	3.00 ± 0.00	4.94e-02 ± 4.73e-02	5.44e-02 ± 2.39e-02	5.09e-02 ± 8.63e-03	7.13e-02 ± 1.09e-02
Osyczka2	3.00 ± 0.00	**1.22e-01 ± 1.35e-01**	**1.18e-01 ± 7.98e-02**	3.00 ± 0.00	1.22e-01 ± 1.35e-01	1.18e-01 ± 7.98e-02	1.74e-01 ± 1.56e-01	1.39e-01 ± 4.08e-02
MED.3D	2.39 ± 0.49	**4.13e-01 ± 1.42e+00**	**5.55e-02 ± 2.44e-02**	3.00 ± 0.00	1.86e+00 ± 6.02e+00	1.17e-01 ± 7.75e-02	1.22e+00 ± 2.10e-01	7.42e-02 ± 5.22e-03
Viennet2	2.38 ± 0.48	6.25e+00 ± 1.37e+01	**6.09e-02 ± 8.44e-03**	3.57 ± 0.49	2.76e+05 ± 4.94e+05	1.94e-01 ± 1.27e-02	**9.37e-01 ± 3.03e-01**	6.72e-02 ± 2.59e-03
MED.5D	2.80 ± 0.40	**2.57e-01 ± 3.33e-01**	**7.93e-02 ± 1.05e-02**	3.00 ± 0.00	7.21e-01 ± 4.12e-01	1.53e-01 ± 1.87e-02	3.33e+00 ± 6.76e-01	1.19e-01 ± 7.53e-03
S3TD	3.25 ± 1.58	2.04e-01 ± 4.04e-02	1.37e-01 ± 3.15e-02	36.20 ± 33.81	5.58e-01 ± 5.99e-01	6.31e-01 ± 6.36e-01	7.17e-01 ± 2.14e-01	9.88e-02 ± 6.67e-03

Results

For each problem and method, the average and the standard deviation of the GD and IGD are shown in Table 1. In Table 1, we highlighted the best score of GD and IGD out of all methods for each problem and added the results of Mann-Whitney's one-tail U-test to check the best score was smaller than that of the other methods significantly. In case of conducting multiple tests, we corrected p-value by Holm's method.

For Bézier simplex fitting methods, the number of iterations until termination is also shown. The table shows that the fitting was finished in approximately three iterations in all cases except for the all-at-once fitting on S3TD.

Inductive skeleton vs. all-at-once For all the two-objective problems, both methods obtained almost identical GD and IGD values. For three- and five-objective problems, the inductive skeleton fitting significantly outperformed the all-at-once fitting in both GD and IGD. Especially in Viennet2, the all-at-once fitting exhibited unstable behavior in GD while the inductive skeleton fitting was stable.

Inductive skeleton vs. response surface The inductive skeleton fitting achieved better IGDs in Schaffer, Osyczka2, 3-MED, and 5-MED but worse values in ConstrEx and Viennet2. In terms of the average GD, the inductive skeleton fitting was better in all the problems except ConstrEx.

Discussion

In this section, we discuss approximation accuracy, required sample size, practicality for real-world data and objective map approximation of our method.

Approximation Accuracy

For three- and five-objective problems, the inductive skeleton fitting achieved slightly better IGDs. The inductive skeleton fitting adjusts a small subset of control points at each step with already-adjusted control points of the faces. This reduces the number of parameters estimated at a time, which seems to prevent over-fitting.

More significant differences can be seen in GD. Figure 4 shows the cause of these defferences: The all-at-once fitting obtained an overly-spreading Bézier simplex while the inductive skeleton fitting found an exactly-spreading Bézier simplex. Minimizing the squared loss (5) only imposes that all sample points are close to the Bézier simplex, which leads to a good IGD. However, it does not impose that all Bézier simplex points are close to the sample points, which is needed to achieve a good GD. The inductive skeleton fitting stipulates that each face of the Bézier simplex must be close to each face of the Pareto front (i.e., the Pareto front of each subproblem), which minimizes GD.

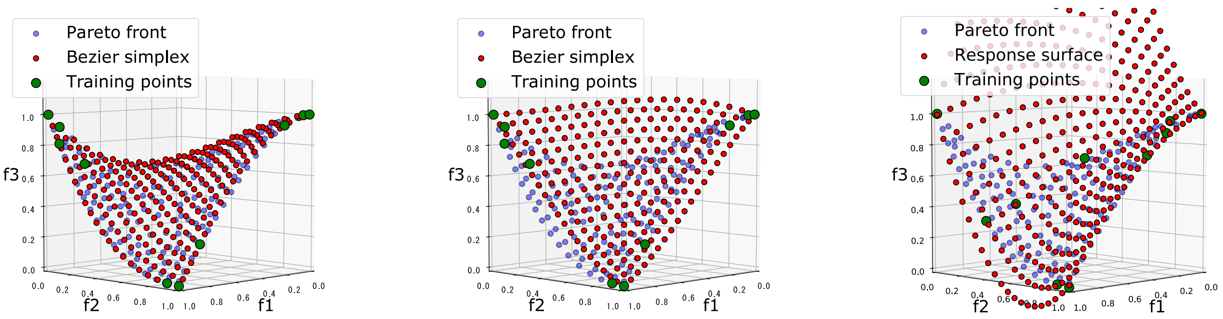
Similarly to the all-at-once fitting, the response surface has poor GDs. As we do not know the true Pareto set, it is difficult for grid sampling to obtain an exactly-spreading surface.

ConstrEx and Viennet2 are the only problems where the inductive skeleton fitting was worse than the response surface method. ConstrEx is a non-smooth curve that cannot be fitted by a single Bézier curve (see Figure 1 in Appendix D). This type of Pareto fronts leads to a challenging problem: developing a method for gluing multiple Bézier simplices to express a non-smooth surface. Viennet2 is a smooth surface but its curvature is severely sharp (see Figure 3 in Appendix D). Although Viennet2 is defined by quadratic functions (see Appendix B), its Pareto front cannot be fitted by the Bézier simplex of degree three. This means that setting the degree of the Bézier simplex grater than or equal to the maximum degree in the problem definition does not ensure approximation accuracy. It would be useful to develop a way to understand the required degree of the Bézier simplex from the problem definition.

Required Sample Size

To achieve good accuracy, how many sample points does the Bézier simplex require? Figure 5 shows transitions of GD and IGD on 5-MED when the sample size of each three-objective subproblem (two-dimensional face) varies $N_3 = 1, \dots, 10$ with fixed $N_1 = 1$ and $N_2 = 2$. Surprisingly, both GD and IGD have already converged at $N_3 = 4$.

In case of $D = 3$ for 5-MED, which is a five-objective ($M = 5$) problem, the number of all control points to be esitimated is $\binom{3+5-1}{5} = 35$. While the all-at-once fitting fits them simultaneously, in case of the induuctive skeleton method, we only have to fit two and one control points for each two- and three-objective subproblem respectively.This result demonstrates that the inductive skeleton fitting allows us to fit Pareto fronts with small samples by reducing the number of control points to be estimated at a time.



(a) Inductive skeleton

(b) All-at-once

(c) Response surface

Figure 4: Bézier triangles for 3-MED with sample size $(1, 2, 1)$.

Practicality for Real-World Data

Now we discuss the practicality of the inductive skeleton fitting. Firstly, we focus on the performance of S3TD, which is a real-world problem. Table 1 indicates that the average IGD of the inductive skeleton fitting on S3TD was worse than that of the response surface method. However, the IGD itself is an order of 10^{-1}, which small in absolute sense considering the number of the objective functions ($M = 4$) and the sample size of the validation set ($N = 58$) on S3TD. The average GD of the inductive skeleton method was smaller than that of the other methods and the difference was significant with significance level $p = 0.1$. Figure 6 shows transitions of GD and IGD on S3TD when the sample size of each three-objective subproblem (two-dimensional face) varies $N_3 = 1, \dots, 10$ with fixed $N_1 = 1$ and $N_2 = 5$. We can observe that both GD and IGD already converged at $N_3 = 4$ as Figure 5. These results suggest that the inductive skeleton fitting can work properly in real-world problems.

In real-world problems, it seems to be difficult to know that the Pareto set/front is a simplex and in some cases, it is degenerated. However, a statistical test has been proposed to judge whether the Pareto set/front of a given problem is a simplex or not (Hamada and Goto 2018). This test tells us whether we can apply the inductive skeleton fitting to real-world problems. Furthermore, even if the Pareto front is degenerated, Theorem 1 states the Bezier simplex model can fit degenerate Pareto sets/fronts.

Objective Map Approximation

As well as the Pareto front approximation discussed and demonstrated so far, Theorem 1 ensures that our method can be used for approximating the Pareto set, $X^*(\boldsymbol{f})$, and the restricted objective map, $\boldsymbol{f} : X^*(\boldsymbol{f}) \to \boldsymbol{f}X^*(\boldsymbol{f})$. Such approximations may provide richer information about the problem. To check the performance of graph approximation, we made a sample of solution-objective pairs of 5-MED $X = \left\{ (\boldsymbol{x}, \boldsymbol{f}(\boldsymbol{x})) \in \mathbb{R}^5 \times \mathbb{R}^5 \;\middle|\; \boldsymbol{x} \in X^*(\boldsymbol{f}) \right\}$ and applied the inductive skeleton fitting to it. Other settings were the same as the Pareto front approximation. Table 2 shows the GD and IGD values where the results of the all-at-once fitting are just for baseline. As the inductive skeleton fitting achieved GD and IGD of 10^{-1} order of magnitude, which means that it accurately approximates the graph of the restricted objective map in the abslute sense. In 5-MED, the graph of the restricted objective map on the Pareto set is a four-dimensional topological simplex in a 10-dimensional space. Although the codimension is six times higher than the case of Pareto front approximation, we have similar GD and IGD to ones of 5-MED in Table 1. This fact implies that the approximation error mainly depends on the intrinsic dimensionality rather than the ambient dimensionality.

Conclusions

In this paper, we have proposed the Bézier simplex model and its fitting algorithm for approximating Pareto fronts of multi-objective optimization problems. An approximation theorem of the Bézier simplex has been proven. Numerical experiments have shown that our fitting algorithm, the inductive skeleton fitting, obtains exactly-spreading Bézier simplices over synthetic Pareto fronts of different shape and dimensionality. It has been also observed that a real-world problem with four objectives but only 58 points is accurately fitted. The proposed model and its fitting algorithm drastically reduce the sample size required to describe the entire Pareto front.

The current algorithm has some drawbacks. The inductive skeleton fitting requires that each subproblem must have a non-empty sample, which may be too demanding for some practice. Furthermore, the loss function for fitting does not take into account of the sampling error of the Pareto front. To get a more stable approximation under wild samples, we plan to extend the (deterministic) Bezier simplex to a probabilistic model.

We believe that the potential applications of the method are not limited to the use in post-optimization process. It will be introduced in evolutionary algorithms and dynamically evolved to accelerate search. We also expect that this model can be applied to regression problems that have complex boundary conditions such as learning from multi-labeled data.

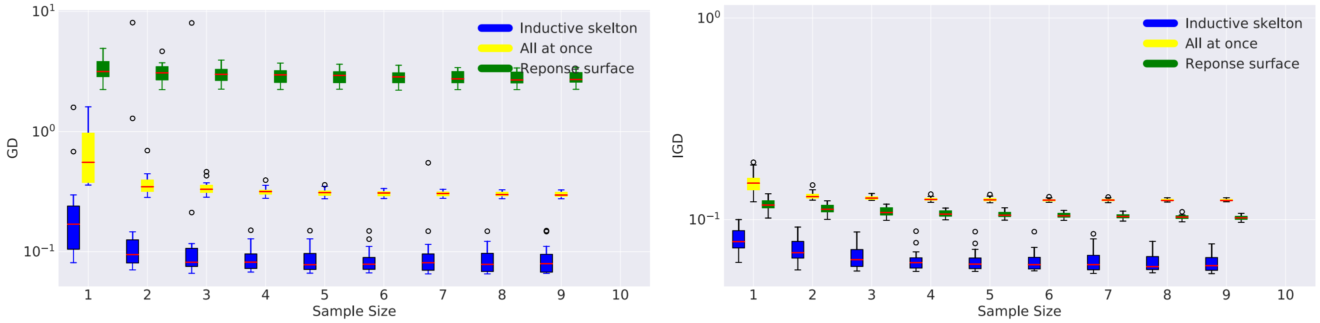


Figure 5: Sample size N_3 vs. GD/IGD on 5-MED with sample size $(1, 2, N_3)$ (boxplot over ten trials).

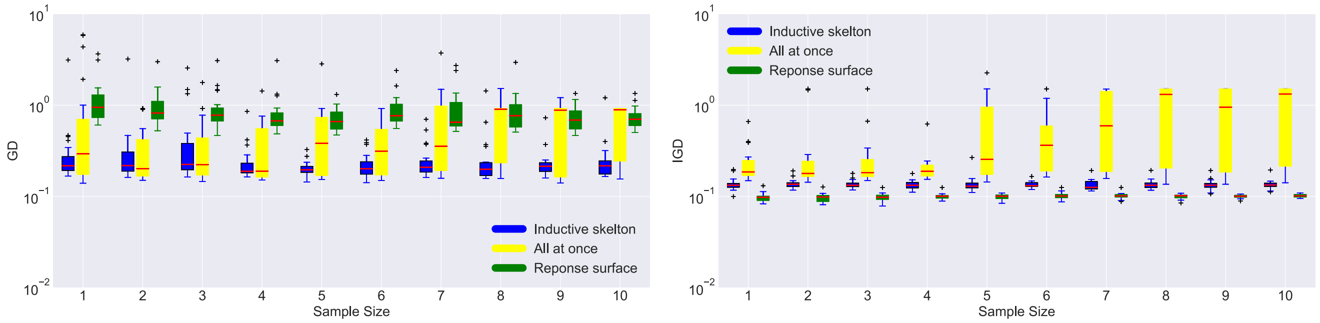


Figure 6: Sample size N_3 vs. GD/IGD on S3TD with sample size $(1, 5, N_3)$ (boxplot over ten trials).

Table 2: GD and IGD (avg. ± s.d. over 20 trials) with sample size $(1, 2, 1)$ for the graph of 5-MED.

	Inductive skeleton			**All-at-once**	
Iteration	GD	IGD	Iteration	GD	IGD
2.85 ± 0.36	3.35e-01 ± 9.27e-02	2.05e-01 ± 3.37e-02	3.00 ± 0.00	1.34e+00 ± 1.95e+00	2.79e-01 ± 8.00e-02

Acknowledgements

We wish to thank Dr. Yuichi Ike for checking the proof and making a number of valuable suggestions.

References

Borges, C. F., and Pastva, T. 2002. Total least squares fitting of Bézier and B-spline curves to ordered data. *Computer Aided Geometric Design* 19(4):275–289.

Chand, S., and Wagner, M. 2015. Evolutionary many-objective optimization: A quick-start guide. *Surveys in Operations Research and Management Science* 20(2):35–42.

Chiba, K.; Makino, Y.; and Takatoya, T. 2009. Design-informatics approach for intimate configuration of silent supersonic technology demonstrator. In *Proceedings of the 47th AIAA Aerospace Sciences Meeting*. Orlando, Florida, USA: American Institute of Aeronautics and Astronautics.

Coello, C. A. C.; Lamont, G. B.; and Van Veldhuisen, D. A. 2007. *Evolutionary Algorithms for Solving Multi-Objective Problems*. Genetic and Evolutionary Computation Series. Springer Science+Business Media, LLC.

Deb, K., and Srinivasan, A. 2006. Innovization: Innovating design principles through optimization. In *Proceedings of the 8th Annual Conference on Genetic and Evolutionary Computation*, GECCO '06, 1629–1636. New York, NY, USA: ACM.

Farin, G. 2002. *Curves and Surfaces for CAGD: A Practical Guide*. Computer graphics and geometric modeling. Morgan Kaufmann.

Gelman, A., and Hill, J. 2007. *Data Analysis Using Regression and Multilevel/Hierarchical Models*. Analytical Methods for Social Research. Cambridge University Press.

Gelman, A.; Carlin, J.; Stern, H.; Dunson, D.; Vehtari, A.; and Rubin, D. 2013. *Bayesian Data Analysis, Third Edition*. Chapman & Hall/CRC Texts in Statistical Science. Taylor & Francis.

Goel, T.; Vaidyanathan, R.; Haftka, R. T.; Shyy, W.; Queipo, N. V.; and Tucker, K. 2007. Response surface approximation of Pareto optimal front in multi-objective optimization. *Computer Methods in Applied Mechanics and Engineering* 196(4):879–893.

Hamada, N., and Goto, K. 2018. Data-driven analysis of Pareto set topology. In *Proceedings of the Genetic and Evolutionary Computation Conference*, GECCO '18, 657–664. New York, NY, USA: ACM.

Hamada, N.; Nagata, Y.; Kobayashi, S.; and Ono, I. 2010. Adaptive weighted aggregation: A multiobjective function optimization framework taking account of spread and evenness of approximate solutions. In *Proceedings of the 2010 IEEE Congress on Evolutionary Computation*, CEC 2010, 787–794.

Kuhn, H. W. 1967. On a pair of dual nonlinear programs. *Nonlinear Programming* 1:38–45.

Li, B.; Li, J.; Tang, K.; and Yao, X. 2015. Many-objective evolutionary algorithms: A survey. *ACM Computing Surveys* 48(1):13:1–13:35.

Lovison, A., and Pecci, F. 2014. Hierarchical stratification of Pareto sets. *ArXiv e-prints*. http://arxiv.org/abs/1407.1755.

Mastroddi, F., and Gemma, S. 2013. Analysis of Pareto frontiers for multidisciplinary design optimization of aircraft. *Aerospace Science and Technology* 28(1):40–55.

Nebro, A.; Durillo, J.; and Vergne, M. 2015. Redesigning the jMetal multi-objective optimization framework. In *Proceedings of the Companion Publication of the 2015 Annual Conference on Genetic and Evolutionary Computation*, GECCO Companion '15, 1093–1100. New York, NY, USA: ACM.

Obayashi, S.; Jeong, S.; and Chiba, K. 2005. Multi-objective design exploration for aerodynamic configurations. In *Proceedings of the 35th AIAA Fluid Dynamics Conference and Exhibit*. Toronto, Ontario, Canada: American Institute of Aeronautics and Astronautics.

Rodríguez-Chía, A. M., and Puerto, J. 2002. Geometrical description of the weakly efficient solution set for multicriteria location problems. *Annals of Operations Research* 111:181–196.

Shao, L., and Zhou, H. 1996. Curve fitting with Bézier cubics. *Graphical Models and Image Processing* 58(3):223–232.

Shoval, O.; Sheftel, H.; Shinar, G.; Hart, Y.; Ramote, O.; Mayo, A.; Dekel, E.; Kavanagh, K.; and Alon, U. 2012. Evolutionary trade-offs, Pareto optimality, and the geometry of phenotype space. *Science* 336(6085):1157–1160.

Smale, S. 1973. Global analysis and economics I: Pareto optimum and a generalization of Morse theory. In Peixoto, M. M., ed., *Dynamical Systems*. Academic Press. 531–544.

Veldhuizen, D. A. V. 1999. *Multiobjective Evolutionary Algorithms: Classifications, Analyses, and New Innovations*. Ph.D. Dissertation, Department of Electrical and Computer Engineering, Graduate School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB, Ohio, USA.

Vrugt, J. A.; Gupta, H. V.; Bastidas, L. A.; Bouten, W.; and Sorooshian, S. 2003. Effective and efficient algorithm for multiobjective optimization of hydrologic models. *Water Resources Research* 39(8):1214–1232.

Zitzler, E.; Thiele, L.; Laumanns, M.; Fonseca, C. M.; and da Fonseca, V. G. 2003. Performance assessment of multiobjective optimizers: An analysis and review. *IEEE Transactions on Evolutionary Computation* 7(2):117–132.