Visualization of Pareto Front Approximations in Evolutionary Multiobjective Optimization: A Critical Review and the Prosection Method

Tea Tušar, *Student Member, IEEE*, and Bogdan Filipič, *Member, IEEE*

***Abstract*—In evolutionary multiobjective optimization, it is very important to be able to visualize approximations of the Pareto front (called approximation sets) that are found by multiobjective evolutionary algorithms. While scatter plots can be used for visualizing 2-D and 3-D approximation sets, more advanced approaches are needed to handle four or more objectives. This paper presents a comprehensive review of the existing visualization methods used in evolutionary multiobjective optimization, showing their outcomes on two novel 4-D benchmark approximation sets. In addition, a visualization method that uses prosection (projection of a section) to visualize 4-D approximation sets is proposed. The method reproduces the shape, range, and distribution of vectors in the observed approximation sets well and can handle multiple large approximation sets while being robust and computationally inexpensive. Even more importantly, for some vectors, the visualization with prosections preserves the Pareto dominance relation and relative closeness to reference points. The method is analyzed theoretically and demonstrated on several approximation sets.**

***Index Terms*—Approximation set, evolutionary algorithm, evolutionary multiobjective optimization, Pareto front, projection, visualization.**

I. Introduction

In evolutionary multiobjective optimization, we wish to simultaneously optimize several (possibly conflicting) objectives. This can be achieved by means of a multiobjective evolutionary algorithm (MOEA), which finds an approximation of the Pareto front, called an approximation set. An approximation set consists of distinct objective vectors that are nondominated with regard to each other, each representing a different trade-off between the objectives. There exist many measures to assess the quality of an approximation set (i.e., how well it approximates the Pareto front in terms of convergence, spread, and distribution of objective vectors) [1], [2]. However, no measure is as effective as the visualization of the approximation set, especially if the Pareto front is known and can be visualized as well.

Manuscript received September 6, 2012; revised July 31, 2013 and November 15, 2013; accepted March 9, 2014. Date of publication March 24, 2014; date of current version March 27, 2015. This work was supported in part by research program P2-0209 and in part by research projects L2-3651 and J2-4120, all funded by the Slovenian Research Agency.

The authors are with the Department of Intelligent Systems, Jožef Stefan Institute, Ljubljana SI-1000, Slovenia, and also with Jožef Stefan International Postgraduate School, Ljubljana SI-1000, Slovenia (e-mail: tea.tusar@ijs.si; bogdan.filipic@ijs.si).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TEVC.2014.2313407

Visualization in evolutionary multiobjective optimization is essential in many aspects—it can be used to [3] estimate the location, range, and shape of the Pareto front, assess conflicts and trade-offs between objectives, select preferred solutions, monitor the progress or convergence of an optimization run, assess the relative performance of different MOEAs, etc. As a prerequisite to accomplish these tasks, a visualization method should be able to preserve the Pareto dominance relation between objective vectors. This means that any relation between objective vectors A and B (A dominates B, B dominates A, or A and B are incomparable) should be evident also from their visualization. This is crucial when comparing two or more approximation sets, since without dominance preservation of a visualized approximation set may seem to dominate another one while this is not the case. Moreover, visualized approximation sets should maintain their shape, range, and distribution of vectors as any large distortion of these features affects our perception of the approximation sets [4]. The shape of the approximation set might be of great importance to the decision maker as it presents the trade-offs among the objectives. Further, a visualization method should be robust, meaning that the addition or removal of a vector within the range of the approximation set should not produce a significantly different visualization. As approximation sets found by MOEAs are often large, a visualization method should be able to handle large sets in terms of visualization capability as well as computational complexity. Additionally, simultaneous visualization of multiple approximation sets is required if different approximation sets are to be compared. Finally, a visualization method should be scalable to multiple dimensions and simple to understand and use.

When tackling optimization problems with two or three objectives, scatter plots have almost all of the mentioned desired properties. See scatter plots of two 2-D and 3-D approximation sets in Fig. 1 (we assume minimization of all objectives). Both clearly show that the two approximation sets are of different shape and range, have different distribution of vectors, and are intertwined (in one region, the linear approximation set dominates the spherical one, while in others the spherical approximation set dominates the linear one). Moreover, scatter plots are simple, robust, and able to visualize a large number of vectors of both sets while being computationally inexpensive. However, their drawback is their poor scalability.

In fact, when the number of objectives m is four or greater, such a simple and intuitive visualization of approximation sets is much harder (if not impossible) to achieve. Since there exists

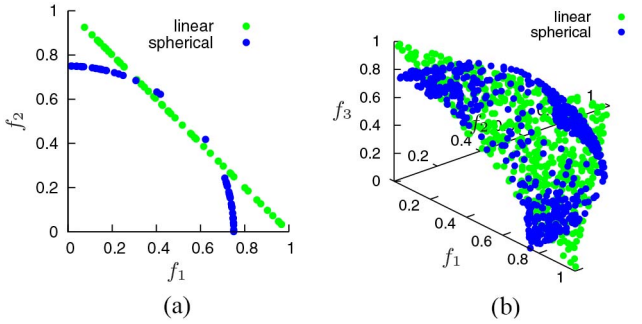

Fig. 1. Scatter plots of (a) 2-D and (b) 3-D benchmark approximation sets.

no general Pareto dominance preserving mapping from a higher-dimensional space to a lower-dimensional space [5], we cannot preserve the Pareto dominance relation between all vectors of a 4-D approximation set. Instead, we can only aspire to preserve the Pareto dominance relation between as many vectors as possible. Similarly, we can only strive to maintain the shape, range, and distribution of vectors as much as possible. When mapping a large number of vectors from four or more dimensions to 2-D or 3-D, robustness and efficiency of this mapping must be assured. Finally, it is very hard to clearly visualize two (or more) large approximation sets simultaneously and scale such a visualization method to any number of objectives.

The task of visualizing approximation sets can be regarded as a multiobjective (or rather, many-objective) optimization problem in its own, in which each of the mentioned requirements needs to be fulfilled as much as possible. Existing visualization methods generally provide solutions to this problem that favor scalability over accuracy (preservation of the dominance relation and other features). We provide a different trade-off with a method that can visualize 4-D approximation sets in 3-D in an intuitive way using prosection (projection of a section). The method is simple, fast, and yet powerful—it is able to preserve the Pareto dominance relation between some vectors as well as their relative closeness to reference points while at the same time reproducing the shape, range, and distribution of vectors in the observed approximation set well. Moreover, it is robust and able to handle multiple large approximation sets. As such it is well suited to comparing different approximation sets. Similarly to scatter plots, its biggest drawback is its poor scalability.

The contribution of this paper is three-fold. First, Section II introduces pairs of BASes that can be used to compare visualization methods, second, the existing methods used for visualizing approximation sets are demonstrated on a pair of 4-D BASes in Section III, and third, the visualization method that uses prosections to visualize 4-D approximation sets is described in detail and illustrated on the BASes as well as other approximation sets in Section IV. The paper concludes with a summary and directions for further work in Section V.

II. Benchmark Approximation Sets

A. Background

Let us first provide formal definitions of the terms from the field of multiobjective optimization that are used in this paper.

The multiobjective optimization problem consists of finding the optimum of a function

$$\boldsymbol{f}\colon X \to F$$
$$\boldsymbol{f}\colon (x_1, \ldots, x_n) \mapsto (f_1(x_1, \ldots, x_n), \ldots, f_m(x_1, \ldots, x_n))$$

where X is an n-dimensional decision space, and F is an m-dimensional objective space ($m \geq 2$). Each solution $\boldsymbol{x} \in X$ is called a decision vector, while the corresponding element $\boldsymbol{f}(\boldsymbol{x}) \in F$ is an objective vector. Without loss of generality we assume that $F \subseteq \mathbb{R}^m$ and all objectives $f_i : X \to \mathbb{R}$ are to be minimized for all $i \in \{1, \ldots, m\}$.

As this paper deals with visualization in the objective space, which can be viewed rather independently from the decision space, the following definitions are confined to the objective space.

Definition 1 (Pareto Dominance Relation of Vectors): The objective vector $\boldsymbol{f}^{\mathrm{A}} = (f_1^{\mathrm{A}}, \ldots, f_m^{\mathrm{A}})$ dominates the objective vector $\boldsymbol{f}^{\mathrm{B}} = (f_1^{\mathrm{B}}, \ldots, f_m^{\mathrm{B}})$, i.e. $\boldsymbol{f}^{\mathrm{A}} \prec \boldsymbol{f}^{\mathrm{B}}$, if

$$f_i^{\mathrm{A}} \leq f_i^{\mathrm{B}} \text{ for } \forall i \in \{1, \ldots, m\} \text{ and } \boldsymbol{f}^{\mathrm{A}} \neq \boldsymbol{f}^{\mathrm{B}}.$$

Definition 2 (Incomparability of Vectors): The objective vectors $\boldsymbol{f}^{\mathrm{A}} = (f_1^{\mathrm{A}}, \ldots, f_m^{\mathrm{A}})$ and $\boldsymbol{f}^{\mathrm{B}} = (f_1^{\mathrm{B}}, \ldots, f_m^{\mathrm{B}})$ are incomparable, i.e. $\boldsymbol{f}^{\mathrm{A}} \parallel \boldsymbol{f}^{\mathrm{B}}$, if

$$\boldsymbol{f}^{\mathrm{A}} \neq \boldsymbol{f}^{\mathrm{B}}, \boldsymbol{f}^{\mathrm{A}} \not\prec \boldsymbol{f}^{\mathrm{B}} \text{ and } \boldsymbol{f}^{\mathrm{B}} \not\prec \boldsymbol{f}^{\mathrm{A}}.$$

Definition 3 (Pareto Optimality): The objective vector $\boldsymbol{f}^*$ is Pareto optimal if there exists no $\boldsymbol{f} \in F$ such that $\boldsymbol{f} \prec \boldsymbol{f}^*$.

When the objectives are conflicting, several different objective vectors can be Pareto optimal. They constitute the Pareto front. The result of a MOEA is usually a set of solutions with mutually incomparable objective vectors. This is called the Pareto front approximation or approximation set for short.

Definition 4 (Approximation Set): A set of objective vectors $\mathcal{A} \subseteq F$ is called an approximation set if $\boldsymbol{f}^{\mathrm{A}} \parallel \boldsymbol{f}^{\mathrm{B}}$ for any two objective vectors $\boldsymbol{f}^{\mathrm{A}}, \boldsymbol{f}^{\mathrm{B}} \in \mathcal{A}$.

The Pareto dominance relation can be defined also on approximation sets.

Definition 5 (Pareto Dominance Relation of Approximation Sets): The approximation set $\mathcal{A}$ dominates the approximation set $\mathcal{B}$, i.e., $\mathcal{A} \prec \mathcal{B}$, if every $\boldsymbol{f}^{\mathrm{B}} \in \mathcal{B}$ is dominated by at least one $\boldsymbol{f}^{\mathrm{A}} \in \mathcal{A}$.

Finally, recall the requirement for a Pareto-dominance preserving mapping.

Definition 6 (Pareto-Dominance Preserving Mapping): The mapping $\Pi : \mathbb{R}^m \to \mathbb{R}^n$, where $n < m$ is a Pareto-dominance preserving mapping, if

$$\boldsymbol{f}^{\mathrm{A}} \prec \boldsymbol{f}^{\mathrm{B}} \iff \Pi(\boldsymbol{f}^{\mathrm{A}}) \prec \Pi(\boldsymbol{f}^{\mathrm{B}})$$

for any two vectors $\boldsymbol{f}^{\mathrm{A}}, \boldsymbol{f}^{\mathrm{B}} \in \mathbb{R}^m$.

B. Comparing Visualization Methods

In the field of evolutionary multiobjective optimization, there exist many benchmark problems (such as, for example, the DTLZ [6] and WFG [7] test suites), which are

used for comparing the performance of MOEAs. However, there exist no benchmark sets that could analogously be used for comparing visualization methods. In fact, no serious attempt to compare visualization methods has been made in this field so far. For this purpose, we introduce the concept of BASes to be used when comparing visualization methods.

Based upon the requirements for visualization methods from the introduction, we can list some specific demands a suite of BASes should conform to. The idea is that BASes should have some distinct properties that can be used when assessing how visualization methods fulfill the aforementioned requirements. As any BAS can consist only of mutually nondominated vectors, BASes in the same suite need to dominate each other entirely or in part. This property is important if we wish to inspect whether the visualization methods manage to (partially) preserve the Pareto dominance relation. Next, to be able to assess the preservation of the shape of approximation sets, BASes should be of different shapes, such as linear, concave, convex, mixed, degenerated, discontinuous, with knees (and possibly others). In order to visualize a different distribution of vectors, BASes in the same suite should have uniform as well as different kinds of nonuniform distributions of vectors. Also, BASes should be scalable to many dimensions to check the scalability of the visualization methods. While no specific requirements are needed to inspect the preservation of the objective range, robustness, and simplicity of the visualization methods, in order to assess their capability to visualize multiple large sets, the BASes should be of a large size—appropriate to their dimensionality.

Following these guidelines, we could come up with a considerable suite of BASes, much like the existing suites of benchmark problems. However, visualization methods cannot be compared as efficiently as optimization algorithms because their outcome on each BAS cannot be measured but must be visualized. Therefore, the size of such a suite of BASes is limited by the number of visualization methods we wish to compare and the amount of space we have to present the results. It is for this reason that the number of BASes to be used throughout this paper is limited to two.

The two BASes will be denoted as linear and spherical according to their shapes. These two (rather simple) shapes were chosen among the others as they appear most often in benchmark problems used in the field of evolutionary multiobjective optimization. The two BASes can be instantiated in any dimension (see Fig. 1 for their 2-D and 3-D instances) and have different distributions and ranges of vectors. In addition, for any dimension m they are intertwined—in one region, the linear BAS dominates the spherical one, while in others, the spherical dominates the linear one. In this paper, we deal with the BASes in four instances of different dimensionality and/or cardinality: 2-D with 50 vectors in each BAS, 3-D with 500 vectors in each BAS, and 4-D with 300 and 3000 vectors in each BAS.[1]

[1]All BASes as well as the approximation sets from Section IV-F can be obtained from http://dis.ijs.si/tea/prosections.htm.

Generation of a vector in the linear BAS

Input: Number of objectives m.

1) Set $g_i = \text{uniformRand}()$ for $i = 1, \ldots, m-1$.
2) Set $g_0 = 0$ and $g_m = 1$.
3) Sort the values $g_i, i = 0, 1, \ldots, m$, so that
$$g_0 \leq g_1 \leq \cdots \leq g_{m-1} \leq g_m.$$
4) Set $f_i = g_i - g_{i-1}$ for all $i = 1, \ldots, m$.

Ouput: The objective vector $(f_1, \ldots, f_m)$.

Fig. 2. Algorithm for generating a vector in the linear BAS. The uniformRand() returns a random value in the interval [0, 1) with a uniform distribution.

C. Linear BAS

The first BAS is linear with all objective vectors satisfying the following constraint:

$$\sum_{i=1}^{m} f_i = 1$$

where each $f_i \in [0, 1]$ and m is the number of objectives. The vectors in the linear BAS are created using the algorithm from Fig. 2 and are uniformly randomly distributed [8].

D. Spherical BAS

The second BAS is spherical with all objective vectors satisfying the following constraint:

$$\sum_{i=1}^{m} f_i^2 = 0.75^2$$

where each $f_i \in [0, 0.75]$ and m is the number of objectives. The vectors in the spherical BAS have a nonuniform distribution—only few vectors are located in the middle of the approximation set, while most of them are near its corners. Therefore, an mD spherical BAS has exactly m regions with a high density of vectors. With this property we try to achieve two goals: 1) a nonuniform distribution of vectors different from the uniform one of the linear BAS and 2) a few almost unconnected regions aimed at mimicking the discontinuous fronts.

Figs. 3 and 4 present the algorithms used to create the spherical BAS. Step 2) of the algorithm from Fig. 3 assures that m regions with a high density of vectors are created. These are then projected on the sphere with radius 0.75 as demonstrated in Step 3). Fig. 4 shows how the nonuniform "U-shaped" distribution is created from the Gaussian distribution.[2]

III. Related Visualization Methods

There exist numerous methods for visualizing multidimensional data, see for example [9], [10]. As we are interested only in methods suitable for visualizing approximation sets, our review is restricted to visualization methods that were previously used for this purpose. We divide these methods into two groups: General and specific, according to their ability

[2]This distribution was preferred to the beta distribution (which is also "U-shaped") because its resulting regions with a high density of vectors are more distinct than those achieved with the beta distribution.

Generation of a vector in the spherical BAS

Input: Number of objectives m.

1) Set $g_i = 0$ for $i = 1, \ldots, m$.
2) While $\sum_{i=1}^{m} g_i^2 > 1$ or ($g_i < 0.5$ for all $i = 1, \ldots, m$) do
   $g_i = \text{nonuniformRand}()$ for each $i = 1, \ldots, m$.
3) Project the vectors on the sphere using:

$$\begin{aligned} f_1 &= 0.75\cos(\phi_1) \\ f_2 &= 0.75\sin(\phi_1)\cos(\phi_2) \\ &\vdots \\ f_{m-1} &= 0.75\sin(\phi_1)\cdots\sin(\phi_{m-2})\cos(\phi_{m-1}) \\ f_m &= 0.75\sin(\phi_1)\cdots\sin(\phi_{m-2})\sin(\phi_{m-1}), \end{aligned}$$

where

$$\begin{aligned} \phi_1 &= \arctan\frac{\sqrt{g_2^2+\cdots+g_m^2}}{g_1} \\ \phi_2 &= \arctan\frac{\sqrt{g_3^2+\cdots+g_m^2}}{g_2} \\ &\vdots \\ \phi_{m-2} &= \arctan\frac{\sqrt{g_{m-1}^2+g_m^2}}{g_{m-2}} \\ \phi_{m-1} &= 2\arctan\frac{g_m}{\sqrt{g_{m-1}^2+g_m^2}+g_{m-1}}. \end{aligned}$$

Ouput: The objective vector $(f_1, \ldots, f_m)$.

Fig. 3. Algorithm for generating a vector in the spherical BAS. The nonuniformRand() function is presented in Fig. 4.

Generation of a random value with a nonuniform distribution

1) Set f to a value outside $[-2, 2]$.
2) While $f < -2$ or $f > 2$ do $f = \text{gaussianRand}()$.
3) If $f \geq 0$ set $f = \frac{f}{4}$, else set $f = 1 + \frac{f}{4}$.

Ouput: The value f.

Fig. 4. Algorithm for generating a random value in the interval [0, 1) with a nonuniform distribution. The gaussianRand() function returns a random number with a normal distribution.

to handle the unique features of approximation sets. For each presented method we provide in Figs. 5 and 6 its visualization of the small 4-D BASes, consisting of 300 vectors each.[3] In addition, Table I summarizes the properties of all methods in view of the requirements for visualization methods presented in the introduction.

The overview of the related methods ends with the presentation of orthogonal prosections as used for visualizing abstract mathematical models.

A. General Multidimensional Data Visualization Methods

The general multidimensional data visualization methods are those introduced outside the field of evolutionary multiobjective optimization and thus make no effort to preserve the Pareto dominance relation between vectors or any other feature specific to evolutionary multiobjective optimization. Therefore, we review these methods mainly with regard to their ability to distinguish between the different shape and distribution of vectors of the two BASes.

1) Scatter Plot Matrix: A straightforward visualization method is to project all vectors to a lower-dimensional space by disregarding all the dimensions of the vector that are beyond those that can be visualized. If this is done for all possible combinations of these lower-dimensional spaces, a scatter plot matrix is obtained [see Fig. 5(a)]. The scatter plot matrix is a very fast, simple, and robust visualization method that in our case retains some information on the shape of the approximation sets (it is easy to distinguish between the spherical and linear BAS) as well as the different distribution of vectors.

2) Bubble Chart: In a scatter plot, additional dimensions can be visualized using size (4-D) and color (5-D), thus obtaining a bubble chart (see [11], [12]; in [13] it is called a trade-off plot). From Fig. 5(b) we can observe that the bubble chart has the same advantages and disadvantages as the scatter plot matrix. The main benefit of the bubble chart over the scatter plot matrix is that all the information is given in a single plot.

3) Radial Coordinate Visualization (RadViz): The idea for RadViz [14] comes from physics. The objectives (called dimensional anchors) are distributed evenly on the circumference of the unit circle. Imagine that each vector is held with springs that are attached to the anchors and the spring force is proportional to the value in the corresponding objective/anchor. The position of the vector is the one where the spring forces are in equilibrium. For example, vectors that are placed close to f_i have a higher value in f_i than in any other objective, while vectors with all equal values are placed exactly in the center of the circle. RadViz was used under the name of barycentric coordinates to visualize approximation sets in [15] and [16]. While the RadViz of our two BASes [see Fig. 5(c)] is able to preserve well the distribution of vectors of both sets, we cannot distinguish their shape.

4) Parallel Coordinates: Using parallel coordinates [17], each m-dimensional vector is represented as a polyline with vertices on the parallel axes, where the position of the vertex on the ith axis corresponds to the ith coordinate of the vector. Parallel coordinates are very useful for representing (in)dependences between objectives. In our case [see Fig. 5(d)], the clutter created by numerous polylines conceals the distribution of vectors, which could be seen otherwise. Although not able to show the shape of approximation sets, parallel coordinates are frequently used for visualizing results in evolutionary multiobjective optimization.

5) Heatmaps: In a heatmap, objective values are shown using color [18]. Similarly to the parallel coordinates plot, heatmaps can show (in)dependences between objectives. See Fig. 5(e), where the vectors in each heatmap are sorted by the value of the first objective. Although with this visualization no information is lost, for our two BASes not much information could be gained either.

The five methods presented so far are very simple to understand and compute—they do not require sophisticated

[3]The original (large) 4-D BASes consist of 3000 vectors each, but only their small subsets are used for these visualizations as the whole sets present a difficulty for most of the methods.

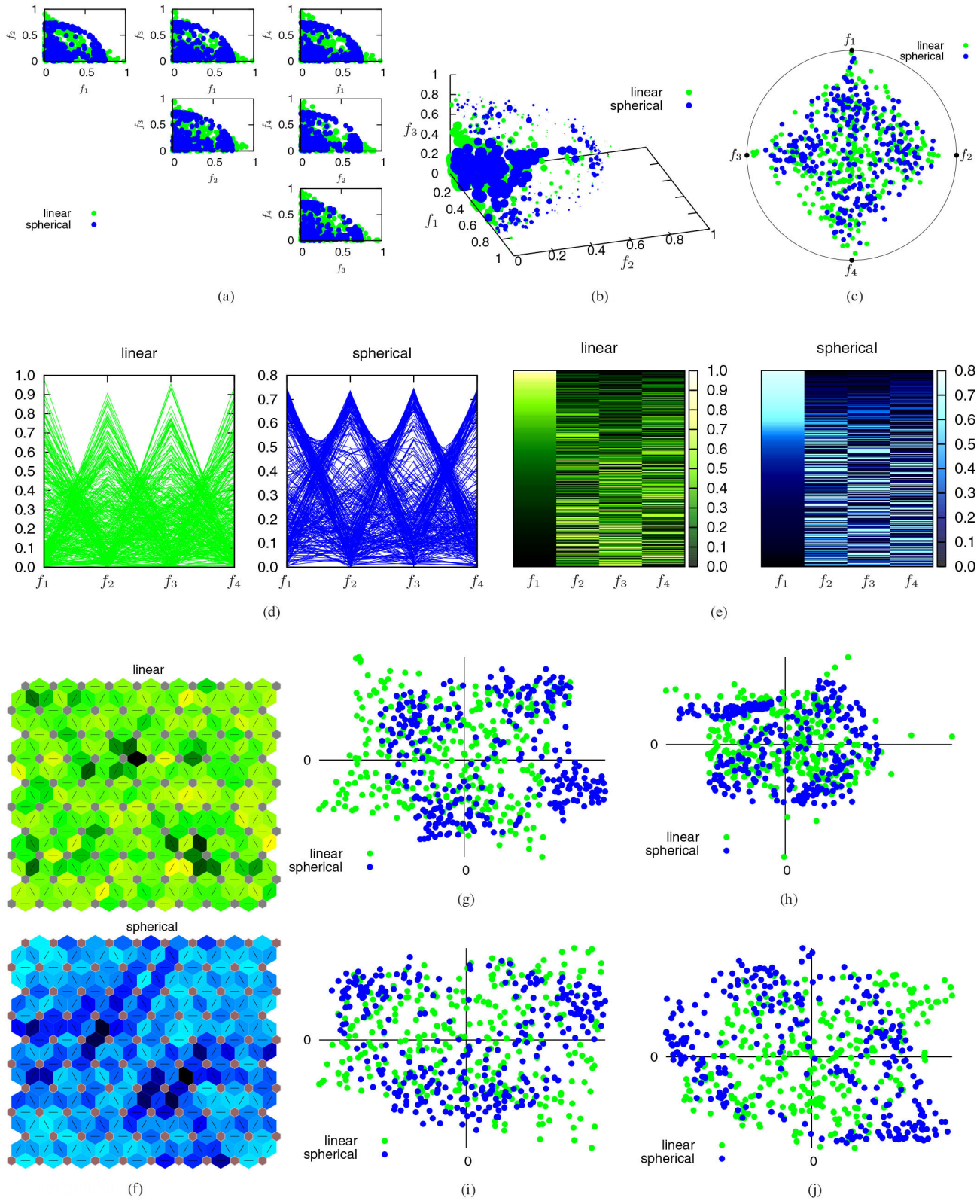

Fig. 5. Visualization of the two small 4-D BASes using general multidimensional data visualization methods (see Section III-A). (a) Scatter plot matrix. (b) Bubble chart. (c) RadViz. (d) Parallel coordinates. (e) Heatmaps. (f) Self organizing maps (SOMs). (g) Sammon mapping. (h) Neuroscale. (i) Principal component analysis (PCA). (j) Isomap.

mappings of vectors—and are therefore very fast. RadViz, parallel coordinates, and heatmaps can also be easily scaled in many dimensions and the latter two are able to visualize the decision space together with the objective space. In our case, the combination of the bubble chart and RadViz provides most information. The next five methods use some more sophisticated mapping to perform dimension reduction to the 2-D space.

6) Sammon Mapping: Sammon mapping [19] aims to minimize the stress function, which emphasizes the preservation of the local distances. This means that distances between the representation of vectors in the visualization space are required to be as close as possible to those in the objective space. The minimization can be performed either by gradient descent, as proposed initially, or by other means, usually involving iterative methods. See [20] for the use of Sammon mapping in evolutionary multiobjective optimization. In our case [Fig. 5(g)] the Sammon mapping preserves very well the distribution of vectors—all four regions with a high density of vectors of the spherical BAS are manifested.

7) Neuroscale: Neuroscale [21] pursues the same goal as Sammon mapping (the preservation of distances) using a radial basis function neural network to minimize the stress function. See [22] for neuroscale representations of an approximation set and Fig. 5(g) for the visualization of our BASes. Neuroscale does not differentiate well between the two BASes. Moreover, it is the only method that skews the distribution of vectors in the linear BAS.

8) Self Organizing Maps (SOMs): The self organizing maps (SOMs) [23] are artificial neural networks that provide a topology preserving mapping from mD to a lower dimension (usually 2-D). This means that nearby vectors in the input space are mapped to nearby units (called neurons) in SOM. While there exist different arrangements of neurons in, we use the hexagonal grid as in [24]. When trained, the SOMs can be visualized using different methods. One of the most popular is the U-matrix (unified distance matrix), in which the distance between adjacent neurons is presented with different colorings. Light areas represent clusters of similar neurons and dark areas indicate cluster boundaries. Fig. 5(f) presents the U-matrices of our two BASes. The SOM of the linear BAS correctly puts all neurons in a single cluster, while it is difficult to interpret the SOM of the spherical BAS (the exact number of clusters is hard to establish).

9) Principal Component Analysis (PCA): Principal component analysis (PCA) finds a new lower-dimensional set of coordinates (the principal components) so that projection onto the principal components captures the maximum variance among all linear projections. The principal components are easily found as the eigenvectors with the highest eigenvalues of the covariance matrix of a set of vectors. PCA was used for visualization in the field of evolutionary multiobjective optimization in [25]. The vectors from our BASes are mapped to the space of the principal components as shown in Fig. 5(i). Two of the regions of high density of vectors in the spherical BAS are visualized as one.

10) Isomap: The basic idea behind Isomap [26] is to preserve the intrinsic geometry of the data when mapping to the 2-D space using multidimensional scaling [27]. In a graph of vectors, in which each vector is linked only to its closest neighbors, the geodesic distance between two vectors is calculated as the sum of Euclidean distances of the shortest path between the two vectors in the graph. The presumption of Isomap is that the vectors lie on some low-dimensional manifold and the distances between vectors along this manifold should be preserved. Isomap was used to visualize approximation sets in [15], while [28] proposes to construct the graph of vectors using distances in the objective space and calculating the geodesic distances in the decision space. As no decision space is given for our BASes, Fig. 5(j) shows the usual Isomap. We can see all four regions with a high density of vectors of the spherical BAS.

The latter five visualization methods are also scalable to many dimensions. However, they are more elaborate, difficult to understand and implement and computationally more expensive than the previous methods. They are also less robust than the previous methods as the mapping used for visualization depends on the values of the objective vectors in the approximation sets. Among these five methods, Sammon mapping was the best to distinguish the distribution of vectors of the two BASes.

B. Methods Specifically Designed for Visualization of Approximation Sets

Here, we review the methods that are tailored to the visualization of multidimensional approximation sets.

1) Distance and Distribution Charts: Reference [29] proposes to plot the vectors from approximation sets against their distance to some approximation of the Pareto front (distance chart) and their distance between each other (distribution chart). As the exact computation of the distribution of vectors is very time consuming when the number of objectives is high, [29] suggests to use a simpler computation that does not produce exact results. Using the latter version, the distance and distribution charts of our BASes are presented in Fig. 6(a). Here, the approximation of the Pareto front consists of nondominated vectors from both sets. The distance chart correctly shows how most of the vectors from the spherical approximation set are dominated, while this holds for only a few vectors from the linear approximation set. However, the distribution chart fails to differentiate between two BASes with a very different distribution of vectors. This might be due to the nonexact computation of the distribution metric, however, despite possible reasons, in our view such charts fail to provide an intuitive presentation of the 4-D BASes.

2) Interactive Decision Maps: In contrast to other methods, the interactive decision maps [30], [31] visualize the Edgeworth–Pareto hull (EPH) instead of the Pareto front (or an approximation of it). The EPH of an approximation set contains all vectors from this set as well as all vectors dominated by this set. This means that instead of visualizing only a finite number of vectors of an approximation set, decision maps visualize a number of axis-aligned sampling surfaces of the EPH. As this is possible only for visualizing 2-D and 3-D approximation sets, interactive choice of the value of the fourth objective is used for visualizing decision maps of 4-D approximation sets. See the example of the decision maps with f_4 fixed to 0.5 of our two BASes in Fig. 6(b). They give a good idea on the shape of the approximation set and also somewhat on the distribution of vectors. However, from the plots it is impossible to infer any dominance relation between the vectors of the two sets.

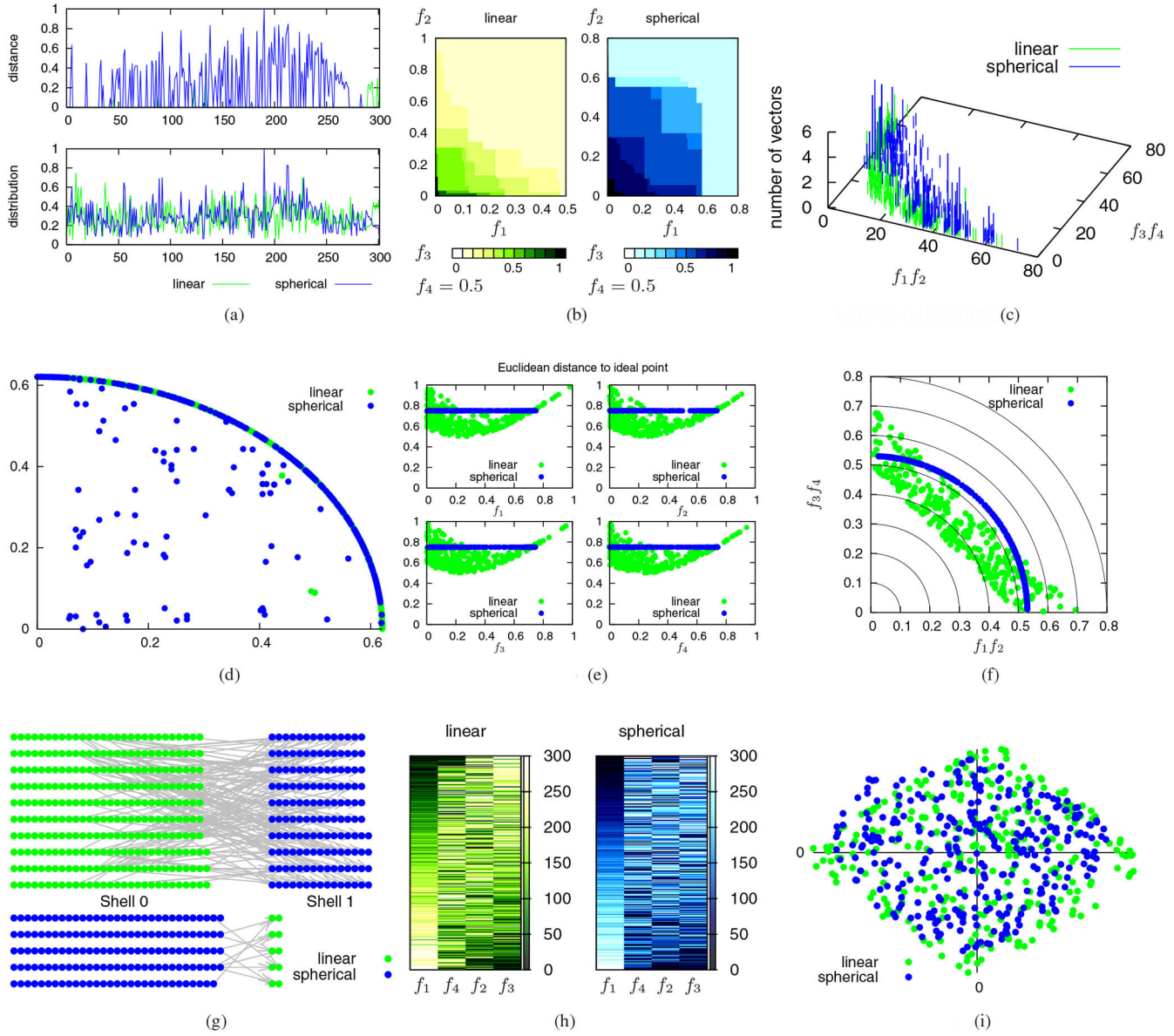

Fig. 6. Visualization of the two small 4-D BASes using methods specific for visualization of approximation sets (see Section III-B). (a) Distance and distribution charts. (b) Interactive decision maps. (c) Hyper-space diagonal counting. (d) Two-stage mapping. (e) Level diagrams. (f) Hyper-radial visualization. (g) Pareto shells. (h) Seriated heatmaps. (i) Multidimensional scaling (MDS).

The task of visualizing the EPH is similar to the task of visualizing the attainment surface [32], which is exactly the surface of the EPH. This was done in [33], but only for the 3-D case.

3) Hyper-Space Diagonal Counting: This method builds upon the idea that the set of natural numbers $\mathbb{N}$ has the same cardinality as the set $\mathbb{N}^m$, where $m \in \mathbb{N}$. Therefore, the set $\mathbb{N}^m$ can be mapped into $\mathbb{N}$ using the hyper-space diagonal counting as described in [34]. Consider now the case of an approximation set in 4-D. Its visualization using hyper-space diagonal counting is performed as follows [35]. First, each objective is divided into a predefined number of bins. The bins of a pair of objectives are then counted using hyper-space diagonal counting producing indices for this pair of objectives. These indices are plotted on two axes (one for each pair of objectives), while the third axis is used to plot the number of vectors of the approximation set that fall in the same set of indices. See Fig. 6(c), where hyper-space diagonal counting is used to visualize the two BASes. Arguably, the plot captures better the distribution of vectors than the shape of the approximation sets. Again, this method does not maintain the dominance relations between vectors.

4) Two-Stage Mapping: The two-stage mapping from [5] aims to preserve (as much as possible) the Pareto dominance and distance relations among vectors. In the first stage, all nondominated vectors are mapped onto on a quarter-circle. A MOEA (in their case, NSGA-II [36]) is used to find a permutation of these vectors so that both relations among vectors (Pareto dominance and distance) are preserved as much as possible. When a good permutation is found, the nondominated vectors are mapped onto the circle in the order given by this permutation (and with distances proportional to their mutual distances). In the second stage, each dominated vector is mapped to the minimal vector of all nondominated vectors

TABLE I
SUMMARY ANALYSIS OF THE VISUALIZATION METHODS WITH REGARD TO THE DESIRABLE PROPERTIES

Method	Preservation[a] of the				Robustness	Handling of large sets	Simultaneous visualization	Scalability	Simplicity
	dominance relation	front shape	objective range	distribution of vectors					
Scatter plot matrix	×	≈	✓	≈	✓	≈[b]	✓	×	✓
Bubble chart	×	≈	✓	≈	✓	≈[b]	✓	×	✓
Radial coord. visual.	×	×	×	≈	✓	≈[b]	✓	✓	✓
Parallel coordinates	≈[c]	×	✓	≈[c]	✓	×	×	✓	✓
Heatmaps	×	×	✓	×	✓	×	×	✓	✓
Sammon mapping	×	×	×	✓	≈	≈[b]	✓	✓	×
Neuroscale	×	×	×	×	≈	≈[b]	✓	✓	×
Self organizing maps	×	×	×	×	≈	✓	×	✓	×
Principal comp. anal.	×	×	×	×	≈	≈[b]	✓	✓	×
Isomap	×	×	×	≈	≈	≈[b]	✓	✓	×
Distan. and distr. ch.	≈	×	×	×	✓	×	✓	≈	×
Interactive dec. maps	×	≈	✓	≈	✓	✓	×	×	✓
Hyper-sp. diag. count.	×	×	×	≈	✓	✓	✓	≈	×
Two-stage mapping	≈	×	×	×	×	≈[b]	✓	≈	×
Level diagrams	×	≈	✓	×	✓	≈[b]	✓	✓	✓
Hyper-radial visual.	×	≈	✓	×	✓	≈[b]	✓	✓	✓
Pareto shells	✓	×	×	×	×	×	✓	✓	✓
Seriated heatmaps	×	×	×	×	≈	×	×	✓	×
Multidim. scaling	×	×	×	×	×	≈[b]	✓	✓	×
Prosections	✓	✓	≈	✓	✓	✓	✓	×	≈

[a] Note that these properties cannot be fully preserved (we are therefore assessing only their partial preservation).
[b] Difficulties with very large sets (containing thousands of vectors).
[c] Possible when only a few vectors are visualized.

that dominate it. Fig. 6(d) shows the result of the two-stage mapping for the two BASes. Unfortunately, other than the split to dominated and nondominated vectors, not much information can be gathered from this plot, while the visualization method is rather complex (requiring itself to solve a multiobjective optimization problem). The two-stage mapping builds its visualization upon the Pareto dominance relations among vectors, which means that in case of addition of a vector to one of the approximation sets (or deletion of a vector from an approximation set), the visualization of the sets might change considerably—depending on how well the inherent multiobjective optimization problem is solved. In other words, this method is not robust.

5) Level Diagrams: Reference [37] proposes to plot the approximation sets on a set of m level diagrams, where m is the number of objectives (the decision space can be visualized using this method, too). In each such diagram, vectors are sorted according to their value of the corresponding objective and plotted against their distance to the ideal point[4] (different norms can be used). Therefore, each vector has the same y position in all diagrams. See the level diagrams of our two BASes in Fig. 6(e), where the Euclidean norm is used for calculating the distance to the ideal point. While the shape of the approximation sets can be inferred from this diagrams, this is not the case for the Pareto dominance relations and the distribution of vectors (particularly for the spherical BAS). Nevertheless, this method is simple, computationally inexpensive and can help the decision maker, especially if color is added to show user preferences.

6) Hyper-Radial Visualization: Somewhat similar to level diagrams is the hyper-radial visualization [38]. Here too the vectors preserve their distance to the ideal point (their hyperradius), but separately for two subsets of objectives. The resulting visualization on our two BASes [see Fig. 6(f)] is able to maintain well the shape of the approximation sets, while the distribution of vectors is correctly represented for the linear BAS, but not for the spherical one. The findings from level diagrams can be applied here too. While the Pareto dominance relations are mostly not preserved, the method is simple, computationally inexpensive and valuable for the decision maker if user preferences are color coded.

7) Pareto Shells: Using the nondominated sorting procedure from NSGA-II, vectors from different approximation sets can be sorted into Pareto shells of mutually nondominated vectors. These shells can be visualized using a graph in which nodes represent vectors (arranged according to the shell they belong to) and directed edges represent the Pareto dominance relation between the connected vectors [39]. Our two BASes are visualized using Pareto shells in Fig. 6(g). While this method is somewhat cumbersome for visualizing large approximation sets (in the plot we only draw one edge for each

[4]For the minimization problem, the *ideal point* is the vector of minimal possible values of this problem in each objective.

dominated vector as drawing all edges would make the plot too crowded), it clearly shows the Pareto dominance relations between vectors. Of course, all other information (objective ranges, distributions of vectors and the shape of the approximation set) cannot be shown using this method. This too is a nonrobust method.

8) Seriated Heatmaps: As the amount of information that can be retrieved from a heatmap heavily depends on the order of vectors in the heatmap, [16] proposes to seriate heatmaps so that similar vectors (and similar objectives) are placed together. Instead of showing actual objective values, seriated heatmaps present ranks that are assigned to each vector component depending on its objective value. The seriated heatmaps for our two BASes are shown in Fig. 6(h). While seriation rearranged the objectives and vectors of both sets, we cannot conclude that seriated heatmaps give us any more information than the regular ones [already presented in Fig. 5(e)]. Note also that because of ranking, seriated heatmaps are not as robust as their predecessors.

9) Multidimensional Scaling (MDS): The classical multidimensional scaling (MDS) [27] tries to find a linear mapping to the 2-D space that preserves similarities between vectors. Simply put, the classical MDS is equivalent to performing PCA on similarities between vectors instead of their distances. In [16], this is done using dominance similarity, which defines two vectors as similar if they dominate the same vectors. Fig. 6(i) shows the MDS of our two BASes using this dominance similarity. Since the dominance similarity takes into account only the relative dominance relations between vectors, the distribution of vectors is lost in such a visualization and the method is even less robust than PCA.

These nine visualization methods are very different from each other and therefore hard to compare. However, in our opinion, the most useful information on our BASes comes from the interactive decision maps and the hyper-radial visualization.

C. Orthogonal Prosections

More than the visualization methods described so far, our method resembles the orthogonal prosections,[5] which were used for visualizing abstract mathematical models [40]. Their idea is very simple (see Fig. 7 for the 3-D case): instead of projecting the whole set of solutions to the orthogonal plane p_1p_2, only the solutions from the chosen section are projected. Because multiple planes can be chosen for the projection (as in the scatter plot matrix), a prosection matrix is used to visualize all orthogonal prosections simultaneously. In addition, color coding can be used for distinguishing between feasible and infeasible solutions.

To our best knowledge, orthogonal prosections were never before used to visualize approximation sets.

IV. Visualization with Prosections

As we have seen in the previous section, there exist numerous ways to visualize 4-D approximation sets. However, none of them can be regarded as a scaled scatter plot with all of its benefits—a clear and informative presentation of the shape, range, and distribution of vectors in the observed approximation sets that preserves the Pareto dominance relation, and the ability of handling multiple large approximation sets while being robust and computationally inexpensive. Moreover, despite all these new visualization possibilities, most researchers in the field of evolutionary multiobjective optimization still resort to parallel coordinates when a 4-D (or higher) approximation set is to be shown. This was our motivation for developing a new visualization method that reduces one dimension of the approximation set using projection of a section and rotation.

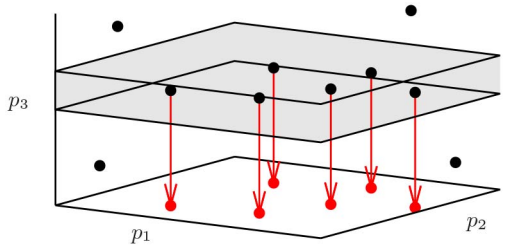

Fig. 7. Orthogonal prosection [40]: a section of p_3 is projected to the orthogonal p_1p_2 plane.

A. Dimension Reduction

As mentioned before, a prosection is a projection of a section (the term was introduced in [41]). Here, the section on the 2-D plane f_1f_2 with origin $\boldsymbol{a} = (a_1, a_2)$ is defined by the angle φ and width d (see [42] for two alternative section definitions). Each vector within the section is orthogonally projected to the line crossing the origin $\boldsymbol{a}$ and intersecting the f_1-axis at angle φ using the mapping $p_{\varphi,d,\boldsymbol{a}}$

$$p_{\varphi,d,\boldsymbol{a}} : (f_1, f_2) \mapsto (f_1', f_2')$$

where

$$f_1' = \cos\varphi((f_1 - a_1)\cos\varphi + (f_2 - a_2)\sin\varphi)$$
$$f_2' = \sin\varphi((f_1 - a_1)\cos\varphi + (f_2 - a_2)\sin\varphi)$$

for all vectors in the section

$$|(f_1 - a_1)\sin\varphi - (f_2 - a_2)\cos\varphi| \leq d.$$

All the vectors in the section are projected to the mentioned line, while all other vectors are ignored [see Fig. 8(a)].

After this mapping, the line with projected vectors needs to be rotated so that this truly becomes a reduction in dimension [see Fig. 8(b)]

$$r : (f_1', f_2') \mapsto \sqrt{(f_1')^2 + (f_2')^2}.$$

When composing these two functions, the transformation simplifies to

$$\begin{aligned} s_{\varphi,d,\boldsymbol{a}}(f_1, f_2) &= r(p_{\varphi,d,\boldsymbol{a}}(f_1, f_2)) \\ &= (f_1 - a_1)\cos\varphi + (f_2 - a_2)\sin\varphi \end{aligned}$$

for all vectors in the section. The function $s_{\varphi,d,\boldsymbol{a}}$ performs dimension reduction from a 2-D to a 1-D space. Let us now show how this can be employed to reduce the mD space to $(m-1)$D.

[5] In [40] they are called prosections, but we added the adjective orthogonal to make a clear distinction between their method and ours.

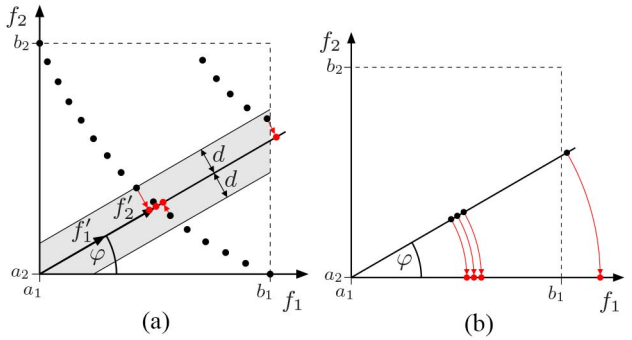

Fig. 8. Graphical presentation of the prosection. (a) Projection. (b) Rotation.

Prosection

Input: Approximation sets in mD, $m \geq 2$.

1) Select the origin $\boldsymbol{a}$ and prosection plane $f_i f_j$, where $i, j \in \{1, \ldots, m\}, i \neq j$.
2) Define the section by choosing the angle φ and section width d. The section contains all vectors $(f_1, \ldots, f_m)$ for which

$$|(f_i - a_i)\sin\varphi - (f_j - a_j)\cos\varphi| \leq d.$$

3) All vectors within the section are projected using the following function:

$$(f_1, \ldots, f_m) \mapsto ((f_i - a_i)\cos\varphi + (f_j - a_j)\sin\varphi, f_{k_1}, \ldots, f_{k_{m-2}})$$

4) All vectors outside the section are ignored.

Ouput: Prosection of the given sets in $(m-1)$D.

Fig. 9. Algorithm for projecting a section of mD approximation sets.

B. Algorithm and Notation

When the number of objectives $m > 2$, other planes beside $f_1 f_2$ are possible. Therefore, we denote with $ijk_1 \ldots k_{m-2}$ a permutation of objective indices $1, \ldots, m$, so that $k_1 < \cdots < k_{m-2}$. The prosection is always performed on the $f_i f_j$ plane, and because of the previous condition, all other objectives are kept in ascending order. The algorithm from Fig. 9 explains step by step how to perform prosection on mD approximation sets.

The prosection affects only two objectives (f_i and f_j) while all the others remain intact and in the same order as before prosection. The new objective that is formed in the prosection (denoted simply by $f_i f_j$) still needs to be minimized, i.e., lower values of $f_i f_j$ are preferred to higher ones.

A prosection of an mD approximation set with origin $\boldsymbol{a}$, prosection plane $f_i f_j$, angle φ, and section width d will be denoted with

$$m\text{D}(\boldsymbol{a}, f_i f_j, \varphi, d)$$

in the rest of the paper.

Note that for any $\varphi \in [0°, 90°]$

$$m\text{D}(\boldsymbol{a}, f_i f_j, \varphi, d) \equiv m\text{D}(\boldsymbol{a}, f_j f_i, (90° - \varphi), d).$$

This means, for example, that the prosection on the plane $f_i f_j$ with angle 30° is equivalent to the prosection on the plane $f_j f_i$ with angle 60°. As a consequence, there is no need to explore both prosections on the plane $f_i f_j$ and on the plane $f_j f_i$.

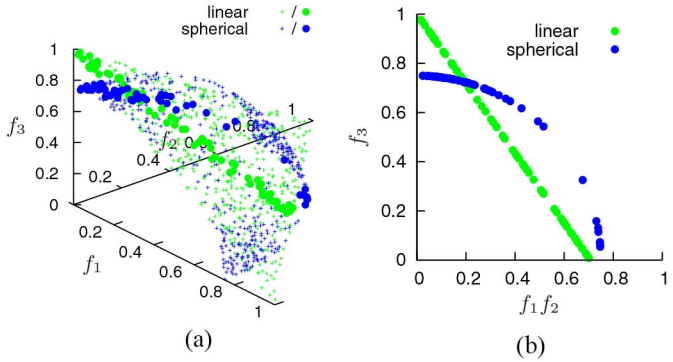

Fig. 10. Visualization with prosections of the 3-D benchmark approximation sets under angle $\varphi = 45°$. (a) Before prosection (vectors in the section are emphasized). (b) After prosection 3-D($\mathbf{0}, f_1 f_2, 45°, 0.05$).

While this transformation can be performed on an mD approximation set for an arbitrary $m \geq 2$, the resulting $(m-1)$D set can be easily visualized only if $m \leq 4$.

C. Visualization of BASes

Let us demonstrate how this method works when projecting 3-D approximation sets to 2-D on the example of the two 3-D BASes from Fig. 1(b). Assume the section is defined by the angle $\varphi = 45°$ and section width $d = 0.05$. This section cuts the plane $f_1 f_2$ in the middle. Fig. 10(a) shows which of the vectors fall in the specified section, while the same vectors after prosection are presented in Fig. 10(b). Essentially, what the method does is slice through approximation set at angle φ and project this slice so that dimension reduction is achieved. Prosections of the 3-D BASes are very similar to the scatter plot of the 2-D BASes [see Fig. 1(a)]. This is desirable since the 3-D BASes are a generalization of the 2-D ones.

In addition, Fig. 11 presents results of prosections under different angles φ. Angles over 45° are not shown as the BASes are nearly symmetric—therefore each 3D($\mathbf{0}, f_1 f_2, \varphi, 0.05$) plot is very similar to the corresponding 3D($\mathbf{0}, f_1 f_2, 90° - \varphi, 0.05$) one. Depending on the angle, the prosections show either one or two regions of the spherical BAS with a high density of vectors. We can see that the range, shape, and distribution of vectors from the visualized part of the objective space are well preserved, while the preservation of the Pareto dominance relation will be discussed in more detail later.

Now, let us focus on the 4-D case. Fig. 12 shows the 4D($\mathbf{0}, f_1 f_2, 45°, 0.25$) visualization with prosection of the small 4-D BASes. As the small BASes contain only 300 vectors each, a larger section ($d = 0.25$) is chosen in order to show the preservation of the linear and spherical shape of the approximation sets. The plot clearly shows the differences in the distribution of vectors from both sets (uniform versus nonuniform distribution). Similar visualizations are achieved also on the large 4-D BASes under different angles φ and with the section width d set to 0.05 (see Fig. 13). Depending on the angle, two or three dense regions of the spherical BAS are visualized. Note again that these prosections resemble very much the 3-D BASes from Fig. 1(b), showing that

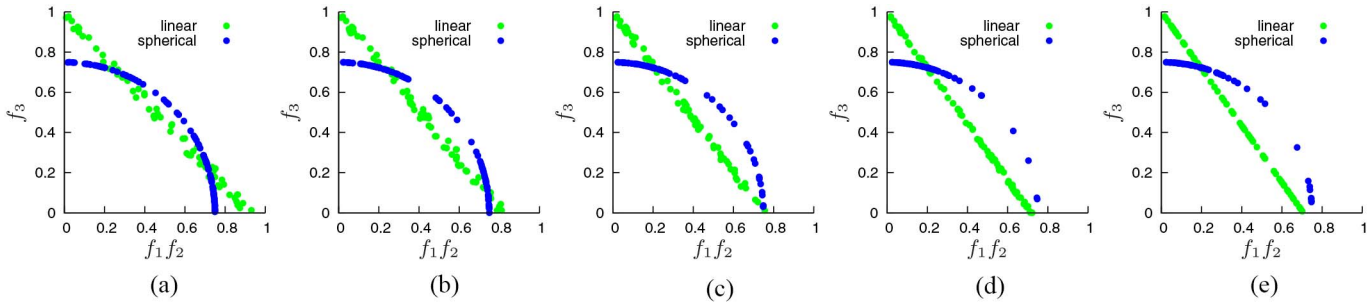

Fig. 11. Prosections of the 3-D benchmark approximation sets under different angles φ. (a) 3D($\mathbf{0}, f_1f_2, 5°, 0.05$). (b) 3D($\mathbf{0}, f_1f_2, 15°, 0.05$). (c) 3D($\mathbf{0}, f_1f_2, 25°, 0.05$). (d) 3D($\mathbf{0}, f_1f_2, 35°, 0.05$). (e) 3D($\mathbf{0}, f_1f_2, 45°, 0.05$).

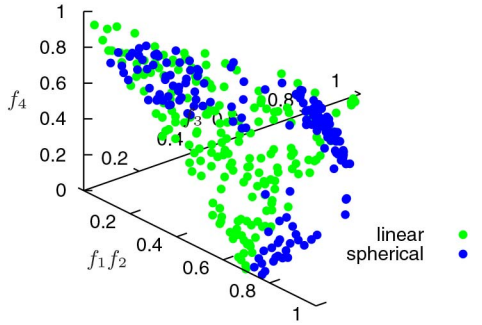

Fig. 12. Prosection 4D($\mathbf{0}, f_1f_2, 45°, 0.25$) of the small 4-D benchmark approximation sets.

they achieve an intuitive visualization of the high dimensional approximation sets.

D. Parameters

The visualization with prosections depends on four parameters: plot origin $\boldsymbol{a}$, prosection plane f_if_j, angle φ, and section width d.

1) Plot Origin and Range of Objectives: For a reasonable result, the plot origin should be set to a vector that dominates all vectors from the approximation sets to be visualized. The ideal point is thus a sensible (but not obligatory) choice for the plot origin. If the origin is chosen to be far better than the ideal point, prosections under extreme angles (near 0° and 90°) and narrow sections might turn out empty.

While the prosection is well defined for any range of objectives, objectives that have ranges of different magnitude affect the "meaning" of the angle φ. For example, the angle $\varphi = 45°$ does not cut the rectangle $[a_i, b_i] \times [a_j, b_j]$ exactly in half if $(b_i - a_i) \neq (b_j - a_j)$. Also, in extreme cases of disproportionate objectives, the size of the section depends heavily on the chosen angle. Therefore, in cases with a big difference between the ranges of objectives, it is best to normalize the objectives prior to visualization.

2) Prosection Plane: In the examples shown so far, we have always performed prosection on a single prosection plane—f_1f_2 (this was not particularly problematic considering the symmetric nature of our BASes). However, in the general case more prosection planes need to be explored to gain a complete "mental picture" of the approximation sets. This can be done simultaneously using a prosection matrix (as in [40]), where each prosection plane results in one plot. As the prosections are symmetric, half of the matrix suffices (as with the scatter plot matrix). Two examples of prosection matrices are provided in Section IV-F. Note also that a chosen prosection plot (or even the whole matrix) can be animated by showing how prosections transform when the angle φ changes. This can further help to construct a mental picture about the trade-off among objectives when the approximation sets are not symmetric.

3) Section Definition: The section is defined with the angle φ and width d. The choice of these two parameters influences greatly the resulting visualization. The angle determines which part of the approximation set is visualized, while the section width regulates the amount of vectors that will be included in the visualization—larger sections produce more crowded plots. See for example the influence of section width in Fig. 14. The section width should be chosen so that it includes enough vectors to visualize the shape of the approximation set and the distribution of vectors (not the case if $d = 0.01$), while at the same time not overcrowding it with too many vectors ($d = 0.25$ is too wide). For our BASes, this means choosing the section width near 0.05 [see Figs. 11(c) and 13(c)].

Fig. 14(b) demonstrates that approximation sets after prosection with a wide section can be indistinct. This depends not only on the width of the section, but also on the chosen angle φ and the shape of the approximation set. We will explain the reasons for this shortly.

E. Properties

The basic difference between the prosections proposed in this paper and the orthogonal prosections presented in Section III-C is in the angle φ, which was either 0° or 90° in previous work (hence we called those prosections orthogonal). The fact that an angle φ different from 0° and 90° is used leads to two important properties of this method, formulated in the next two theorems (their proofs are in the Appendix).

Theorem 1: Suppose the mD($\boldsymbol{a}, f_if_j, \varphi, d$) prosection is performed, where $m \geq 2$ and $\varphi \in (0°, 90°)$. Then for any two vectors $\boldsymbol{f}^{\mathrm{A}} = (f_1^{\mathrm{A}}, \ldots, f_m^{\mathrm{A}})$, and $\boldsymbol{f}^{\mathrm{B}} = (f_1^{\mathrm{B}}, \ldots, f_m^{\mathrm{B}})$ inside the section the following holds. If

$$\boldsymbol{f}^{\mathrm{A}} \prec \boldsymbol{f}^{\mathrm{B}}$$

then

$$m\mathrm{D}(\boldsymbol{a}, f_if_j, \varphi, d)(\boldsymbol{f}^{\mathrm{A}}) \prec m\mathrm{D}(\boldsymbol{a}, f_if_j, \varphi, d)(\boldsymbol{f}^{\mathrm{B}}).$$

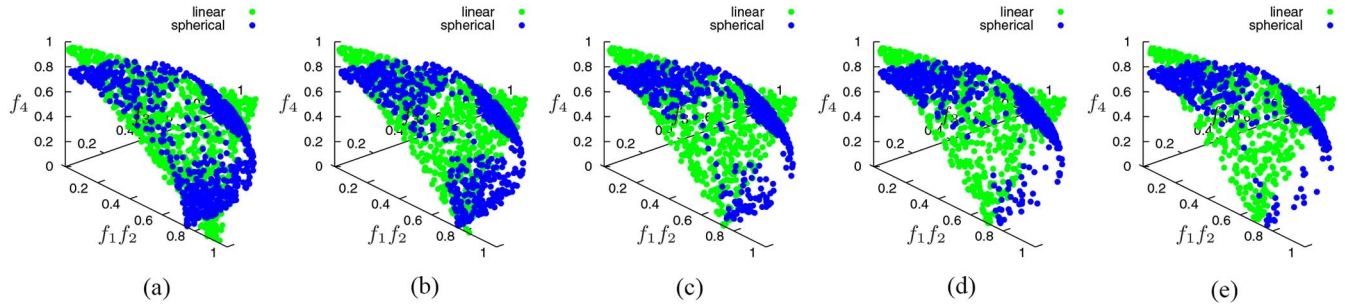

Fig. 13. Prosections of the large 4-D benchmark approximation sets under different angles φ. (a) 4D($\mathbf{0}, f_1f_2, 5°, 0.05$). (b) 4D($\mathbf{0}, f_1f_2, 15°, 0.05$). (c) 4D($\mathbf{0}, f_1f_2, 25°, 0.05$). (d) 4D($\mathbf{0}, f_1f_2, 35°, 0.05$). (e) 4D($\mathbf{0}, f_1f_2, 45°, 0.05$).

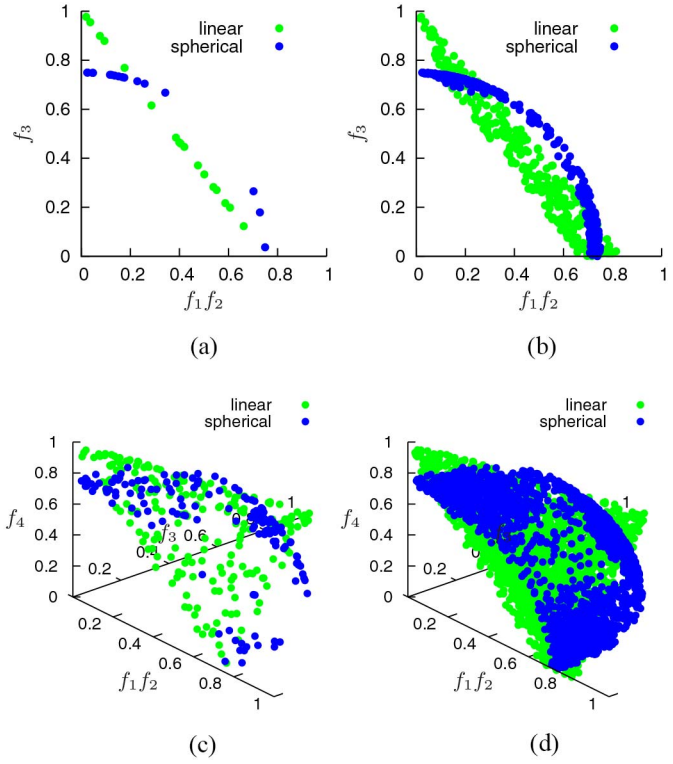

Fig. 14. Prosections of the 3-D and the large 4-D BASEs using different values for section width d ($d = 0.01$ for the left hand side plots and $d = 0.25$ for the right hand side plots). (a) 3D($\mathbf{0}, f_1f_2, 25°, 0.01$). (b) 3D($\mathbf{0}, f_1f_2, 25°, 0.25$). (c) 4D($\mathbf{0}, f_1f_2, 25°, 0.01$). (d) 4D($\mathbf{0}, f_1f_2, 25°, 0.25$).

This means that if one vector dominates the other, the dominance relation is retained after prosection. While it is beneficial that a visualization method is capable of correctly showing the dominance relations among vectors, the other way around (being able to infer the dominance relations from the visualization) is even more important for the correct understanding of the visualized approximation sets.

As shown in [5], no Pareto-dominance preserving mapping exists.[6] Nevertheless, for prosections we can prove that if one projected vector dominates another projected vector and the two are apart enough, the first vector indeed dominates the second one.

[6]It is easy to see why the right-to-left direction from Definition 6 does not hold for prosections. If one vector does not dominate the other, after prosection the first projected vector might dominate the second projected vector.

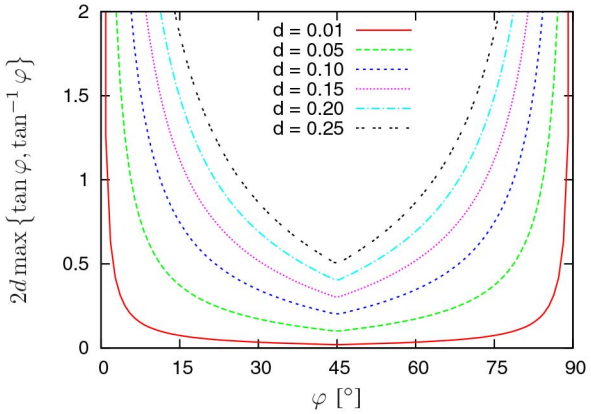

Fig. 15. Values of $2d \max\left\{\tan\varphi, \tan^{-1}\varphi\right\}$ depending on the angle φ and section width d.

Theorem 2: Suppose the mD($\boldsymbol{a}, f_if_j, \varphi, d$) prosection is performed, where $m \geq 2$ and $\varphi \in (0°, 90°)$. Then for any two vectors $\boldsymbol{f}^{\mathrm{A}} = (f_1^{\mathrm{A}}, \ldots, f_m^{\mathrm{A}})$, and $\boldsymbol{f}^{\mathrm{B}} = (f_1^{\mathrm{B}}, \ldots, f_m^{\mathrm{B}})$ inside the section the following holds. If

$$m\mathrm{D}(\boldsymbol{a}, f_if_j, \varphi, d)(\boldsymbol{f}^{\mathrm{A}}) \prec m\mathrm{D}(\boldsymbol{a}, f_if_j, \varphi, d)(\boldsymbol{f}^{\mathrm{B}})$$

and

$$s(f_i^{\mathrm{B}}, f_j^{\mathrm{B}}) - s(f_i^{\mathrm{A}}, f_j^{\mathrm{A}}) \geq 2d \max\left\{\tan\varphi, \tan^{-1}\varphi\right\}$$

then

$$\boldsymbol{f}^{\mathrm{A}} \prec \boldsymbol{f}^{\mathrm{B}}.$$

"Apart enough" is thus any distance in the new objective f_if_j that is greater than $2d \max\left\{\tan\varphi, \tan^{-1}\varphi\right\}$. Unfortunately, $\max\left\{\tan\varphi, \tan^{-1}\varphi\right\}$ strongly depends on the chosen angle φ. If $\varphi = 45°$, then $\tan\varphi = \tan^{-1}\varphi = 1$, which is the smallest possible value. For all other values of φ, the value of $\max\left\{\tan\varphi, \tan^{-1}\varphi\right\}$ is higher, getting unpractical high values in the proximity of $\varphi = 0°$ and $\varphi = 90°$. Fig. 15 shows how the value of $2d \max\left\{\tan\varphi, \tan^{-1}\varphi\right\}$ depends on the values of φ and d.

Theorem 2 tells us that we cannot completely trust the visualized dominance relations. While some vectors may appear (non)dominated in the prosection plot, this might not be the case in the original objective space. Only those vectors that are dominated according to Theorem 2 are truly dominated, while for the rest we cannot be sure. Fig. 16 shows on the example

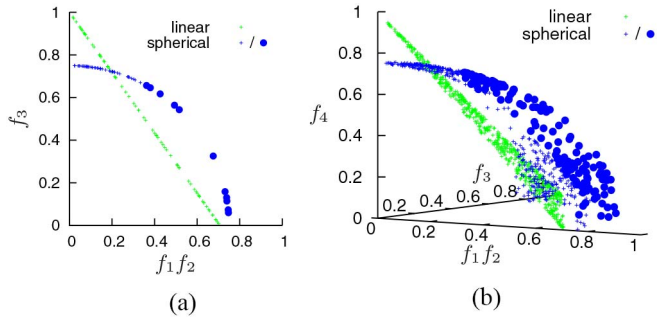

Fig. 16. Prosections of the 3-D and the large 4-D BASes. Vectors from the spherical BASes that are dominated by the vectors from the linear BASes according to Theorem 2 are emphasized (drawn using dots instead of small crosses). (a) 3D$(\mathbf{0}, f_1 f_2, 45°, 0.05)$. (b) 4D$(\mathbf{0}, f_1 f_2, 45°, 0.05)$.

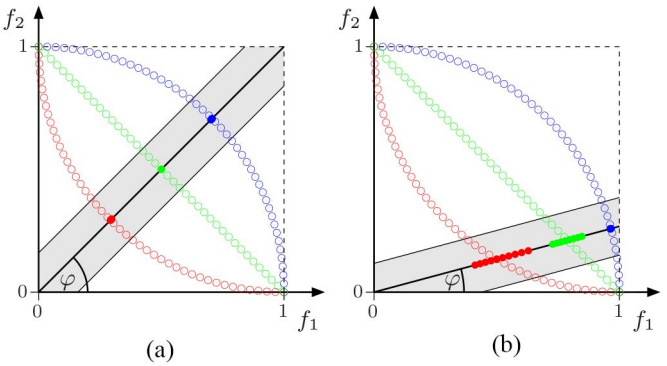

Fig. 17. Prosections of 2-D convex, linear, and concave approximation sets using different values for angle φ. (a) $\varphi = 45°$. (b) $\varphi = 15°$.

of our BASes which vectors from the spherical BASes are dominated by the vectors from the linear BASes according to Theorem 2.

Additionally, Theorem 2 explains the indistinctness mentioned before, as $2d \max\left\{\tan\varphi, \tan^{-1}\varphi\right\}$ is exactly the maximum possible width of indistinctness that an approximation set can achieve in the new objective. The actual indistinctness of an approximation set after prosection depends on its shape—see the example of three 2-D approximation sets of different shape (convex, linear, and concave) in Fig. 17. When $\varphi = 45°$, all approximation sets after prosection are distinct, since they are all almost perpendicular to the section. On the other hand, after prosection with $\varphi = 15°$, the concave approximation is still distinct, but this is not the case for the linear (some indistinctness) and the convex (a lot of indistinctness) ones.

Note that none of these two theorems is true if φ is equal to 0° or 90°, which means that simple orthogonal prosections do not share these useful properties.

Next, let us explore the interpretation of the new objective $f_i f_j$. Because of the projection, much of the information on f_i and f_j is lost, but not all. Assume the mD$(\boldsymbol{a}, f_i f_j, \varphi, d)$ prosection is performed and we are interested in the original values in objectives f_i and f_j of the projected vector with value A in objective $f_i f_j$. Then, we know that the original values of f_i and f_j lie on the line segment $\overline{A'A''}$, where

$$A' = (a_i + A\cos\varphi - d\sin\varphi, a_j + A\sin\varphi + d\cos\varphi)$$
$$A'' = (a_i + A\cos\varphi + d\sin\varphi, a_j + A\sin\varphi - d\cos\varphi).$$

For example, if we are interested in the point A with value 0.5 in objective $f_1 f_2$ of the prosection 4D$(\mathbf{0}, f_1 f_2, 45°, 0.05)$, we know that the original values in f_i and f_j lie on the line segment between points $(0.318, 0.389)$ and $(0.389, 0.318)$.

Finally, a note on the transformation of distances. As a projection from an mD space to a $(m-1)$D space is performed, the distances among arbitrary vectors cannot be preserved. However, it is trivial to show that the distance between two vectors after prosection is never greater than the distance between the original vectors. The distance is preserved when the line segment bounded by the two original vectors is parallel to the line intersecting the plane $f_i f_j$ at the angle φ.

Theorem 3: Suppose the mD$(\boldsymbol{a}, f_i f_j, \varphi, d)$ prosection is performed, where $m \geq 2$. Then for any two vectors $\boldsymbol{f}^{\mathrm{A}} = (f_1^{\mathrm{A}}, \ldots, f_m^{\mathrm{A}})$ and $\boldsymbol{f}^{\mathrm{B}} = (f_1^{\mathrm{B}}, \ldots, f_m^{\mathrm{B}})$ inside the section

$$\left\| m\mathrm{D}(\boldsymbol{a}, f_i f_j, \varphi, d)(\boldsymbol{f}^{\mathrm{A}}) - m\mathrm{D}(\boldsymbol{a}, f_i f_j, \varphi, d)(\boldsymbol{f}^{\mathrm{B}}) \right\| \leq \leq \left\| \boldsymbol{f}^{\mathrm{A}} - \boldsymbol{f}^{\mathrm{B}} \right\|.$$

The equality holds iff

$$\frac{f_j^{\mathrm{A}} - f_j^{\mathrm{B}}}{f_i^{\mathrm{A}} - f_i^{\mathrm{B}}} = \tan\varphi.$$

More importantly, we are able to show that prosections preserve the relative closeness to the reference point for some vectors. This is especially useful when (one or more) reference points are given and we wish to visualize them together with the approximation set.

Theorem 4: Suppose the mD$(\boldsymbol{a}, f_i f_j, \varphi, d)$ prosection is performed, where $m \geq 2$. Let $\boldsymbol{f}^{\mathrm{A}} = (f_1^{\mathrm{A}}, \ldots, f_m^{\mathrm{A}})$, $\boldsymbol{f}^{\mathrm{B}} = (f_1^{\mathrm{B}}, \ldots, f_m^{\mathrm{B}})$ and $\boldsymbol{f}^{\mathrm{R}} = (f_1^{\mathrm{R}}, \ldots, f_m^{\mathrm{R}})$ be three vectors inside the section and let us assume that $\left\|\boldsymbol{f}^{\mathrm{A}} - \boldsymbol{f}^{\mathrm{R}}\right\| < \left\|\boldsymbol{f}^{\mathrm{B}} - \boldsymbol{f}^{\mathrm{R}}\right\|$. If $\left\|\boldsymbol{f}^{\mathrm{B}} - \boldsymbol{f}^{\mathrm{R}}\right\|^2 - \left\|\boldsymbol{f}^{\mathrm{A}} - \boldsymbol{f}^{\mathrm{R}}\right\|^2 > 4D^2$, then

$$\left\| m\mathrm{D}(\boldsymbol{a}, f_i f_j, \varphi, d)(\boldsymbol{f}^{\mathrm{A}}) - m\mathrm{D}(\boldsymbol{a}, f_i f_j, \varphi, d)(\boldsymbol{f}^{\mathrm{R}}) \right\| < \left\| m\mathrm{D}(\boldsymbol{a}, f_i f_j, \varphi, d)(\boldsymbol{f}^{\mathrm{B}}) - m\mathrm{D}(\boldsymbol{a}, f_i f_j, \varphi, d)(\boldsymbol{f}^{\mathrm{R}}) \right\|.$$

On the other hand, if

$$\left\| m\mathrm{D}(\boldsymbol{a}, f_i f_j, \varphi, d)(\boldsymbol{f}^{\mathrm{B}}) - m\mathrm{D}(\boldsymbol{a}, f_i f_j, \varphi, d)(\boldsymbol{f}^{\mathrm{R}}) \right\|^2 - - \left\| m\mathrm{D}(\boldsymbol{a}, f_i f_j, \varphi, d)(\boldsymbol{f}^{\mathrm{A}}) - m\mathrm{D}(\boldsymbol{a}, f_i f_j, \varphi, d)(\boldsymbol{f}^{\mathrm{R}}) \right\|^2 > 4d^2$$

then $\left\|\boldsymbol{f}^{\mathrm{A}} - \boldsymbol{f}^{\mathrm{R}}\right\| < \left\|\boldsymbol{f}^{\mathrm{B}} - \boldsymbol{f}^{\mathrm{R}}\right\|$.

According to Theorem 4, if vectors $\boldsymbol{f}^{\mathrm{A}}$ and $\boldsymbol{f}^{\mathrm{B}}$, where $\boldsymbol{f}^{\mathrm{A}}$ is closer to the reference point $\boldsymbol{f}^{\mathrm{R}}$ than $\boldsymbol{f}^{\mathrm{B}}$, are "apart enough" from the perspective of the reference point ($\|\boldsymbol{f}^{\mathrm{B}} - \boldsymbol{f}^{\mathrm{R}}\|^2 - \|\boldsymbol{f}^{\mathrm{A}} - \boldsymbol{f}^{\mathrm{R}}\|^2 > 4d^2$), the vector $\boldsymbol{f}^{\mathrm{A}}$ remains closest to the reference point also after prosection. And vice versa, from the distances between the vectors and the reference point after prosection we can infer on their closeness in the original space. This means that prosections are able to truthfully visualize the closeness to the reference point for some vectors.

Note that the properties described in the latter two theorems are independent on the chosen angle φ. The proofs to both theorems can be found in the Appendix.

F. Usage Examples

Here, we show how prosections can be used for visualizing: 1) solutions to problems with redundant objectives; 2) various shapes of Pareto fronts (different from the spherical and linear used until now); and 3) the progress of a MOEA.

1) Visualization in Case of Redundant Objectives: Since our approach slices through approximation sets, we wish to explore how this affects visualization of solutions to problems with redundant objectives. An objective is redundant if its elimination does not affect the Pareto front of the given problem. The DTLZ5(I, M) problem family [43], [44] presents problems with redundant objectives, where I denotes the dimension of the Pareto front, and M is equal to the number of objectives. Here, we use two 4-D problems, DTLZ5(3, 4) and DTLZ5(2, 4), which have one and two redundant objectives, respectively, and are defined with

$$\begin{aligned}
\min f_1(\boldsymbol{x}) &= (1+g(\boldsymbol{x}))\cos(\theta_1(\boldsymbol{x}))\cos(\theta_2(\boldsymbol{x}))\cos(\theta_3(\boldsymbol{x}))\\
\min f_2(\boldsymbol{x}) &= (1+g(\boldsymbol{x}))\cos(\theta_1(\boldsymbol{x}))\cos(\theta_2(\boldsymbol{x}))\sin(\theta_3(\boldsymbol{x}))\\
\min f_3(\boldsymbol{x}) &= (1+g(\boldsymbol{x}))\cos(\theta_1(\boldsymbol{x}))\sin(\theta_2(\boldsymbol{x}))\\
\min f_4(\boldsymbol{x}) &= (1+g(\boldsymbol{x}))\sin(\theta_1(\boldsymbol{x}))\\
g(\boldsymbol{x}) &= \sum_{i=4}^{n}(x_i-0.5)^2\\
\theta_i(\boldsymbol{x}) &= \begin{cases} \frac{\pi}{2}x_i & \text{if } i=1,\ldots,I-1\\ \frac{\pi}{4(1+g(\boldsymbol{x}))}(1+2g(\boldsymbol{x})x_i) & \text{if } i=I,\ldots,3\end{cases}\\
0 &\le x_i \le 1, i=1,2,\ldots,n
\end{aligned}$$

where n is the number of decision variables and additional constraints apply. The DTLZ5(3, 4) problem has two constraints

$$\begin{aligned}
f_4(\boldsymbol{x})^2+f_3(\boldsymbol{x})^2+2f_1(\boldsymbol{x})^2 &\ge 1\\
f_4(\boldsymbol{x})^2+f_3(\boldsymbol{x})^2+2f_2(\boldsymbol{x})^2 &\ge 1
\end{aligned}$$

while the DTLZ5(2, 4) problem has three

$$\begin{aligned}
f_4(\boldsymbol{x})^2+4f_1(\boldsymbol{x})^2 &\ge 1\\
f_4(\boldsymbol{x})^2+4f_2(\boldsymbol{x})^2 &\ge 1\\
f_4(\boldsymbol{x})^2+2f_3(\boldsymbol{x})^2 &\ge 1.
\end{aligned}$$

In the DTLZ5(3, 4) problem, the Pareto front is characterized by $2f_1^2+f_3^2+f_4^2=1$ and $f_1=f_2$, which means that either of the first two objectives is redundant. In the DTLZ5(2, 4) problem, vectors on the Pareto front comply to $4f_1^2+f_4^2=1$ and $f_1=f_2=\frac{\sqrt{2}}{2}f_3$, meaning that two among the first three objectives are redundant. This implies that the Pareto front is a surface in the first case and a curve in the second one.

We approximate the two Pareto fronts with two approximation sets, each consisting of 3000 vectors, and visualize these sets with a prosection matrix, which enables us to view the prosections on all prosection planes simultaneously. Fig. 18 shows the prosection matrix using angle $\varphi=45°$ and width $d=0.05$ for all prosection planes. Because using prosections only slices of approximation sets are visualized, we would expect the plots to show only a small part of each approximation set at a time. They mostly do—with the exception of the first one on the prosection plane f_1f_2, which coincidentally (because for all vectors $f_1=f_2$, i.e., all vectors lie on the hyperspace that intersects f_1f_2 under the angle $\varphi=45°$) shows the whole approximation sets.

It is interesting to inquire whether the existence of redundant objectives could be inferred solely from visualization with prosections. In problems with two redundant objectives, this should be possible. If the Pareto front is a curve, no prosection will visualize it as a surface. In the other case, the problem has only one redundant objective and might not be so straightforward. As we have seen from Fig. 18, some specific views might visualize the whole surface. However, if for a chosen prosection plane, the approximation set is visualized as a strip of a surface regardless of the angle, we could speculate on the existence of a single redundant objective.

2) Visualizing the Shape of Pareto Fronts: To show how prosections visualize approximation sets with a shape different from the spherical and linear, two different multiobjective problems with known Pareto fronts are used: the first is WFG1 from the WFG test problem toolkit [7] and the second is DEB4DK, a 4-D version of the DEB3DK test problem [45].

a) The WFG1 Test Problem: The WFG test problem toolkit can be used to create scalable multiobjective test problems with different characteristics [7]. In this example, we use the 4-D WFG1 test problem which has an interesting "mixed" convex and concave shape of the Pareto front. The front is sampled using 3000 vectors. The objectives of this problem have different ranges—for the vectors on the Pareto front the following holds: $f_i \in [0, 2i]$, $i=1,\ldots,4$. Therefore, we normalize all vectors in the approximation set to lie in $[0, 1]^4$ prior to visualization. The normalized approximation set is visualized with the prosection matrix in Fig. 19.

Depending on the prosection plane, the visualization is capable of showing the mixed convex and concave shape of the Pareto front (when f_4 is not included in the prosection plane—the left-hand side and central plots) or not (when f_4 is included in the prosection plane—the right-hand side plots). This is reasonable, since in the first case we "slice through the waves," while in the second one, we "slice along them."

This example shows that it is indeed important to visualize more than just a single prosection to gain a full understanding of a non symmetric approximation set. Alternatively, additional information on the approximation set can be gained through animation of a chosen prosection plot by changing the angle φ. For example, when the prosection $4\mathrm{D}(\mathbf{0}, f_3f_4, \varphi, 0.05)$ is animated by changing φ from 0° to 90° using the step 5°, we can see that the approximation set "oscillates" (repeatedly comes closer to the plot origin and then draws away from it). This indicates that the approximation set is "wavy," which is a property of the set that cannot be otherwise easily seen using only the prosection plane f_3f_4.

b) The DEB4DK Test Problem: This problem has *knees*—regions on the Pareto front where a small improvement in one objective leads to a large deterioration in at least one other objective. Knees are especially important for decision-making purposes as they are usually preferred to other parts of the Pareto front. It is, therefore, important to be able to show them when visualizing an approximation set with knees.

The first problems with knees have been defined in [45], where the DEB2DK and DEB3DK test problems have two

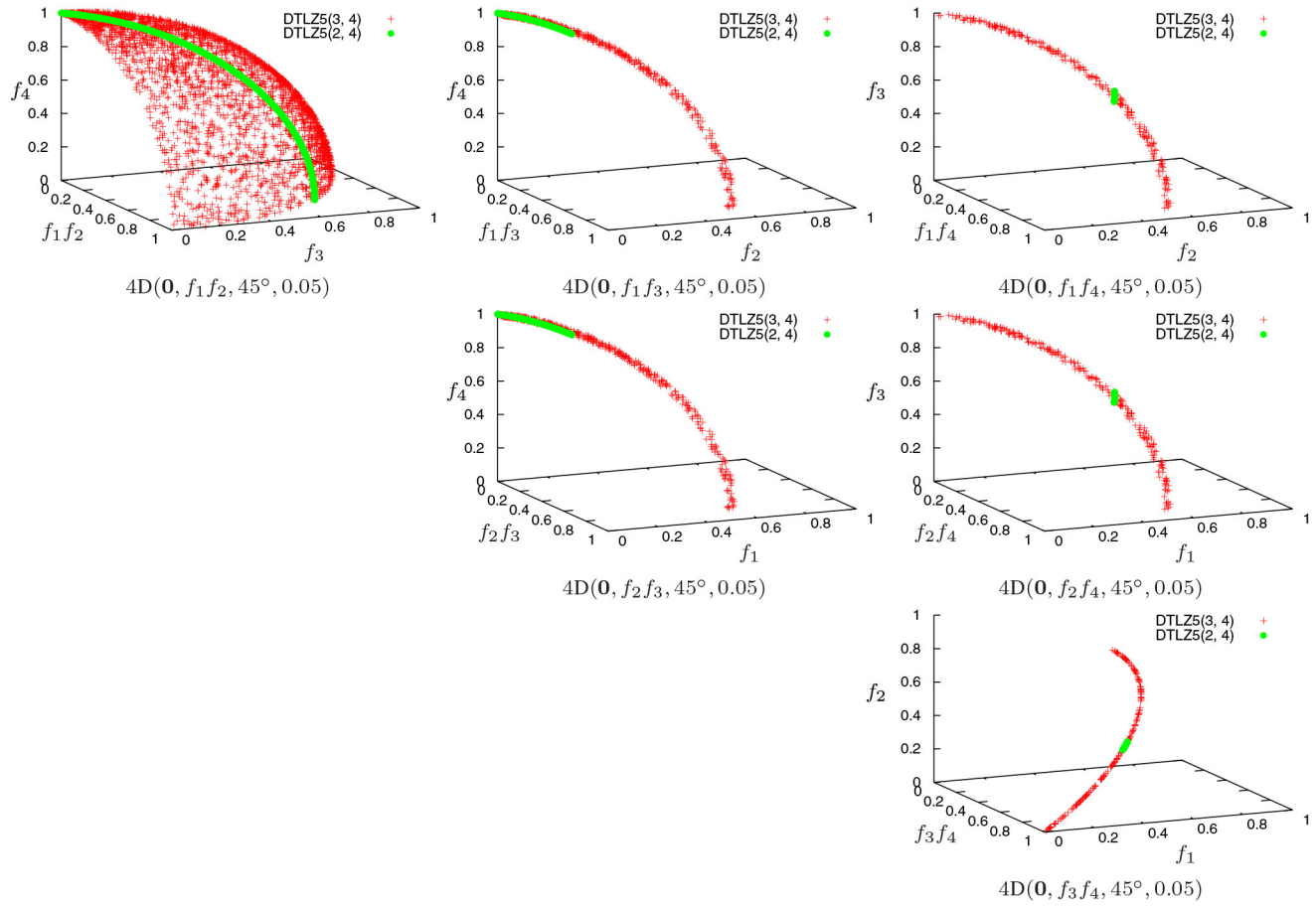

Fig. 18. Prosection matrix of the approximation sets of the DTLZ5(3, 4) and DTLZ5(2, 4) test problems with redundant objectives.

and three objectives, respectively. For example, the DEB3DK test problem is defined as

$$\begin{aligned}
\min f_1(\boldsymbol{x}) &= g(\boldsymbol{x}) r(\boldsymbol{x}) \sin\left(\frac{\pi}{2}x_1\right) \sin\left(\frac{\pi}{2}x_2\right) \\
\min f_2(\boldsymbol{x}) &= g(\boldsymbol{x}) r(\boldsymbol{x}) \sin\left(\frac{\pi}{2}x_1\right) \cos\left(\frac{\pi}{2}x_2\right) \\
\min f_3(\boldsymbol{x}) &= g(\boldsymbol{x}) r(\boldsymbol{x}) \cos\left(\frac{\pi}{2}x_1\right) \\
g(\boldsymbol{x}) &= 1 + \frac{9}{n-1}\sum_{i=2}^{n} x_i \\
r(\boldsymbol{x}) &= \frac{r_1(x_1) + r_2(x_2)}{2} \\
r_i(x_i) &= 5 + 10(x_i - 0.5)^2 + \frac{2\cos(2K\pi x_i)}{K} \\
0 &\le x_i \le 1, i = 1, 2, \ldots, n.
\end{aligned}$$

Here, n is the number of dimensions of the decision space and K is a parameter that together with the number of objectives m determines the number of knees in the Pareto front K^{m-1}. We show the approximation set consisting of 500 vectors from the Pareto front of the DEB3DK problem with $K = 1$ in Fig. 20(a). Because $K = 1$, this Pareto front has only one knee, which is clearly visible from the plot of the approximation set.

Since DEB2DK and DEB3DK are based on the DTLZ problems, they are scalable to any number of objectives. However, they haven't been scaled to more than 3-D by their authors. Here, we introduce the 4-D version of this problem (we call it DEB4DK)

$$\begin{aligned}
\min f_1(\boldsymbol{x}) &= g(\boldsymbol{x}) r(\boldsymbol{x}) \sin\left(\frac{\pi}{2}x_1\right) \sin\left(\frac{\pi}{2}x_2\right) \sin\left(\frac{\pi}{2}x_3\right) \\
\min f_2(\boldsymbol{x}) &= g(\boldsymbol{x}) r(\boldsymbol{x}) \sin\left(\frac{\pi}{2}x_1\right) \sin\left(\frac{\pi}{2}x_2\right) \cos\left(\frac{\pi}{2}x_3\right) \\
\min f_3(\boldsymbol{x}) &= g(\boldsymbol{x}) r(\boldsymbol{x}) \sin\left(\frac{\pi}{2}x_1\right) \cos\left(\frac{\pi}{2}x_2\right) \\
\min f_4(\boldsymbol{x}) &= g(\boldsymbol{x}) r(\boldsymbol{x}) \cos\left(\frac{\pi}{2}x_1\right) \\
g(\boldsymbol{x}) &= 1 + \frac{9}{n-1}\sum_{i=2}^{n} x_i \\
r(\boldsymbol{x}) &= \frac{r_1(x_1) + r_2(x_2) + r_3(x_3)}{3} \\
r_i(x_i) &= 5 + 10(x_i - 0.5)^2 + \frac{3\cos(2K\pi x_i)}{K} \\
0 &\le x_i \le 1, i = 1, 2, \ldots, n.
\end{aligned}$$

Again, we use the DEB4DK problem with $K = 1$, which means that the Pareto front of this problem has only one knee, too. The Pareto front is again sampled with 3000 vectors, but normalization is not needed since all objectives have similar ranges.[7] Note however, that larger objective values require a

[7]The ranges of objectives of the DEB3DK and DEB4DK problems in this paper differ from the ones presented in [45]. This might be due to an unwanted integer division in the original implementation of these two problems (more specifically, in the calculation of the $g(\boldsymbol{x})$ function).

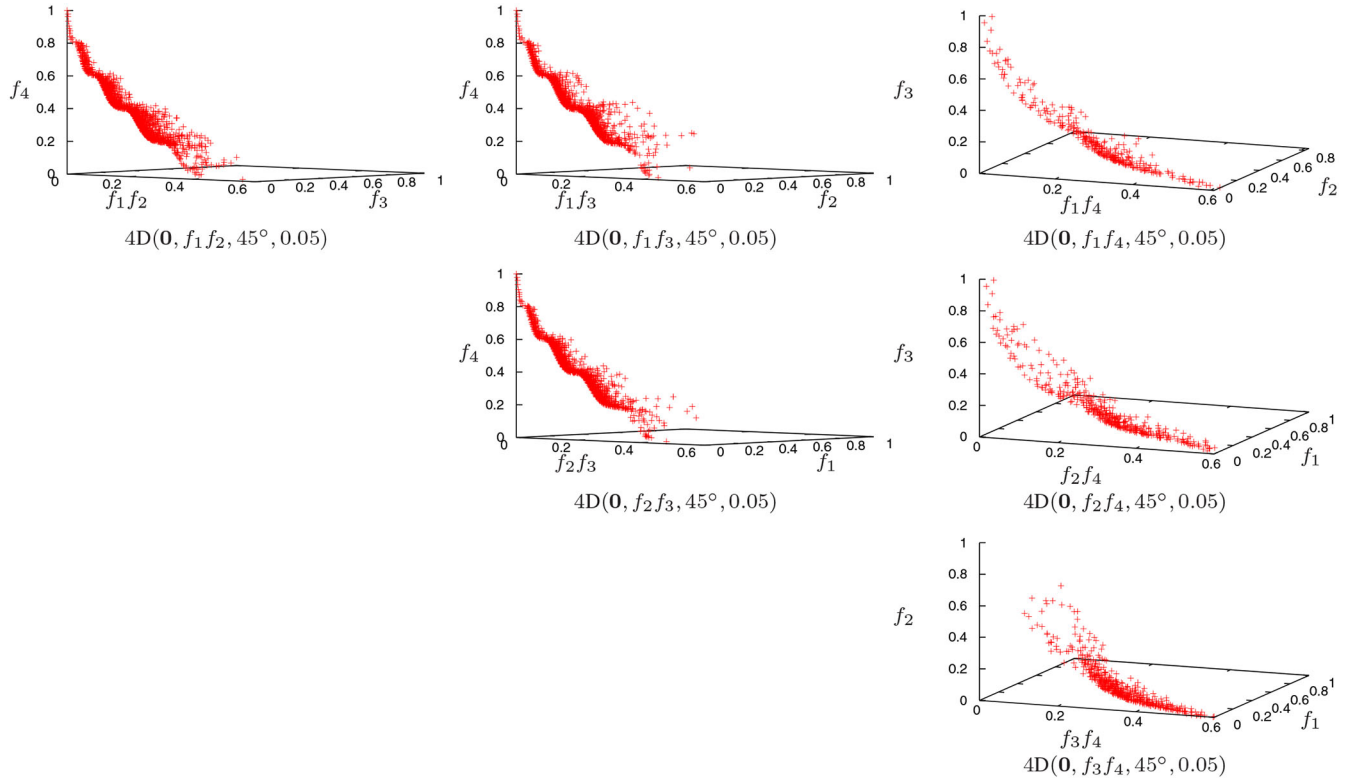

Fig. 19. Prosection matrix of the approximation sets of the WFG1 test problem.

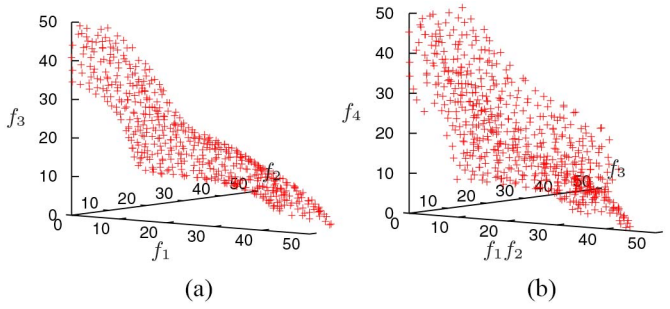

Fig. 20. Approximation sets of the DEB3DK and DEB4DK test problems. (a) 3-D. (b) $4D(\mathbf{0}, f_1f_2, 45°, 2.5)$.

larger section width than usual (we choose $d = 0.05 \times 50 = 2.5$). Fig. 20(b) presents the visualization of this approximation set using prosections. We show only the visualization on one prosection plane as the others produce very similar results. We can see that under the angle 45° the knee is nicely visualized.

In summary, the WFG1 and DEB4DK test problems have shown the potential of prosections for visualizing the shape of Pareto fronts.

3) Visualizing the Progress of MOEA: So far we have used only "artificial" approximation sets that were not achieved as a result of a MOEA. Therefore, in this last usage example we wish to show how prosections can be used to visualize the progress of a true MOEA. To this end we use the differential evolution for multiobjective optimization (DEMO) algorithm [46] on the DTLZ7 benchmark optimization problem with four objectives [6]. The Pareto front of this problem has eight disconnected regions and disproportionate objective values ($f_1, f_2, f_3 \in [0, 1]$ while $f_4 \in [2.9, 8]$), therefore normalization is required for the fourth objective.

The DEMO algorithm with the population of 100 vectors was run on this problem. To visualize the progress of DEMO, we plot the approximation sets achieved by the algorithm after 50, 100, and 300 generations, which have 329, 1022, and 3616 vectors, respectively (see Fig. 21).

Again, we show only the visualizations on two prosection planes as the objectives f_1, f_2, and f_3 are symmetric and produce very similar plots. The first prosection [using f_1f_2, see Fig. 21(a)] shows four different regions. It is easy to see that with increasing generations the algorithm was able to converge better. Note also that the increasing number of vectors in the approximation sets is properly visualized. While the second prosection [using f_3f_4, see Fig. 21(b)] exhibits similar characteristics, it is interestingly able to show five regions simultaneously.

G. Discussion

We can look at visualization with prosections in view of the desired properties for a visualization method (see Introduction and Table I). As shown in Section IV-E (and proven in the Appendix), visualization with prosections is able to preserve the Pareto dominance relation and relative closeness to reference points for some vectors, which is crucial for the correct interpretation of the visualized results in the decision-making process following optimization. In addition, all the visualized approximation sets have demonstrated that this method is also good at maintaining their shape characteristics (for example, knees) and distribution of vectors. However, because the prosection is done under an angle, the ranges of the two objectives

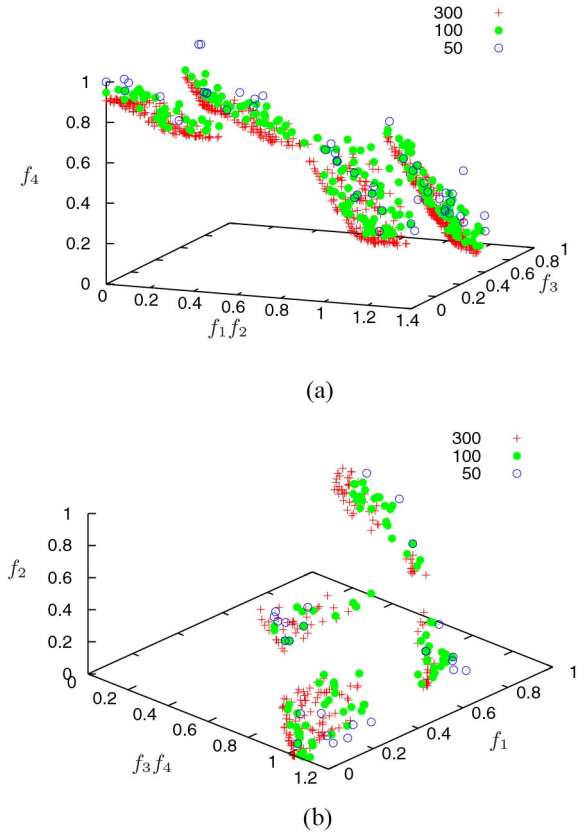

Fig. 21. Prosections of the approximation sets at 50, 100, and 300 generations of the DEMO$^{\text{NS-II}}$ algorithm on the DTLZ7 problem. (a) 4D($\mathbf{0}, f_1 f_2, 45°, 0.05$). (b) 4D($\mathbf{0}, f_3 f_4, 45°, 0.05$).

included in the prosection lose some of their meaning (see Section IV-E).

Further, prosections are as robust as scatter plots—addition or removal of a vector does not importantly change the visualization. Because only a part of the approximation set is visualized at a time, they are able to visualize large approximation sets. Moreover, they are computational inexpensive as they require only a simple mapping to be performed on the chosen two objectives.

One of the best properties of prosections is that they are able to visualize two (or more) approximation sets simultaneously, therefore allowing for direct comparison between different approximation sets. This means that they can be used to study the quality of convergence to the Pareto front (if known), to compare different MOEAs and visualize the progress of a single MOEA (as we have done in Section IV-F3).

On the other hand, their biggest disadvantage in their present form is that they are limited to 4-D approximation sets. Scalability to five or more dimensions is challenging for two reasons. First, applying prosection multiple times might result in a loss of ability to preserve the Pareto dominance relation and maintain the shape, range, and distribution of vectors, which is essential for producing meaningful visualizations. Second, because only a small part of the objective space is visualized at a time, this would scale too, meaning that it would become impractical to show enough views to gain a good understanding of the approximation set. Handling these challenges is a task for future work.

Finally, a note on whether prosections are simple to understand and use. The idea behind prosections is very simple—they slice through the 4-D approximation set and visualize this slice in 3-D. However, in practice some parameters need to be set (see Section IV-D) and several prosection planes and angles should be explored to gain a full understanding of an approximation set. This makes them less simple to use. To enhance their usability, we suggest to follow this procedure for visualization of a 4-D approximation set, in which all four objectives need to be minimized.[8]

1) If the objectives are disproportionate, normalize the approximation sets to the interval $[0, 1]^4$ (for the sake of brevity we assume the approximation sets are normalized from this point on).
2) Set the point $\mathbf{0}$ as the origin and choose d depending on the size of the approximation sets (for example, $d \in [0.02, 0.05]$ for approximation sets with 3000 vectors—d can be smaller for larger sets and larger for smaller sets).
3) Look at the prosection matrix at different angles (for example $\varphi = 0°, 10°, \ldots, 90°$) either with separate plots or animation.
4) Choose the prosection plane and angle φ that give most information and visualize and analyze only this one.
5) Repeat the previous step if needed.

Using this procedure the user gets a small number of meaningful visualizations that can help understand better the 4-D approximation sets.

V. Conclusion

Visualization of Pareto front approximations has different requirements than visualization of other multidimensional data. We are interested not only in the distribution of vectors in the objective space but also in the dominance relations between them (to be able to compare different approximation sets) and in the shape of approximation sets (we wish to see their knees, discontinuities, etc.). Moreover, visualization methods need to handle large approximations sets as the sets found by MOEAs are usually large. To inspect how visualization methods comply with these requirements we have introduced two novel 4-D BASes. They are close together in the objective space, but have a different shape and distribution of vectors. Therefore, a good visualization method should be able to recognize their features and differentiate well between them.

We have shown the visualizations of the existing methods on the two BASes and summarized their properties in a table. Most of the presented methods are scalable to any number of objectives, but fail to correctly show the dominance relations between vectors, the approximation set shape or the distribution of vectors. Moreover, some are not suitable for comparing two or more approximation sets and face difficulties when visualizing large sets.

[8] A few `gnuplot` scripts to support this visualization procedure can be found at http://dis.ijs.si/tea/prosections.htm

The presented visualization with prosections has just the opposite properties: it is able to correctly show the dominance relations between many vectors, the approximation set shape and the distribution of vectors, but is not easily scalable to more than 4-D. In addition, it can handle multiple large approximation sets while being robust and computationally inexpensive. Because of this, prosections are best used to study the quality of convergence to the Pareto front (if known), to compare different MOEAs and visualize the progress of a single MOEA. This was demonstrated on some well-known multiobjective optimization problems as well a problem with knees and problems with redundant objectives, while the partial preservation of the dominance relation and the relative closeness to reference points was formally proven.

This paper tackled visualization in a somehow limited scope as only four objectives were considered. However, in our opinion, we should be able to first understand and really see the 4-D approximation sets before moving on to more than four dimensions. Also, while the step from 3-D to 4-D might seem small, it is very important. Most MOEAs that work well in 2-D and 3-D fail to reach good results in 4-D. Being able to visualize their outcome on 4-D problems would give researchers a powerful tool for finding pitfalls and improving the performance of these algorithms on 4-D optimization problems.

Nevertheless, we are interested in exploring how prosections can be extended beyond 4-D. While at the first glance this seems straightforward—just apply prosection twice—we wish to find a way that will retain all the good properties of 4-D prosections while at the same time not introduce new parameters to keep the method as simple and manageable as possible. One way to do this would be to construct some kind of a "recommendation function" that would provide a ranking of views with regard to their importance to the user. This ranking would be based on the properties of the approximation sets to be visualized. In this way, the user would not have to set any parameters of the method or look at the prosection matrix, but would simply visualize the top recommended views.

APPENDIX
PROOFS OF THEOREMS

In the proofs, we will denote with $ijk_1 \dots k_{m-2}$ the permutation of indices $1, \dots, m$ so that $k_1 < \dots < k_{m-2}$ and use the following abbreviations:

$$\begin{aligned} s^{\mathrm{A}} &:= s_{\varphi,d,\boldsymbol{a}}(f_i^{\mathrm{A}}, f_j^{\mathrm{A}}) = (f_i^{\mathrm{A}} - a_i)\cos\varphi + (f_j^{\mathrm{A}} - a_j)\sin\varphi \\ s^{\mathrm{B}} &:= s_{\varphi,d,\boldsymbol{a}}(f_i^{\mathrm{B}}, f_j^{\mathrm{B}}) = (f_i^{\mathrm{B}} - a_i)\cos\varphi + (f_j^{\mathrm{B}} - a_j)\sin\varphi \\ \sigma^{\mathrm{A}} &:= m\mathrm{D}(\boldsymbol{a}, f_i f_j, \varphi, d)(\boldsymbol{f}^{\mathrm{A}}) = (s^{\mathrm{A}}, f_{k_1}^{\mathrm{A}}, \dots, f_{k_{m-2}}^{\mathrm{A}}) \\ \sigma^{\mathrm{B}} &:= m\mathrm{D}(\boldsymbol{a}, f_i f_j, \varphi, d)(\boldsymbol{f}^{\mathrm{B}}) = (s^{\mathrm{B}}, f_{k_1}^{\mathrm{B}}, \dots, f_{k_{m-2}}^{\mathrm{B}}). \end{aligned}$$

Definition 7 (Weak Pareto Dominance Relation of Vectors): The objective vector $\boldsymbol{f}^{\mathrm{A}} = (f_1^{\mathrm{A}}, \dots, f_m^{\mathrm{A}})$ weakly dominates the objective vector $\boldsymbol{f}^{\mathrm{B}} = (f_1^{\mathrm{B}}, \dots, f_m^{\mathrm{B}})$, i.e. $\boldsymbol{f}^{\mathrm{A}} \preceq \boldsymbol{f}^{\mathrm{B}}$, if

$$f_i^{\mathrm{A}} \leq f_i^{\mathrm{B}} \text{ for } \forall i \in \{1, \dots, m\}.$$

Theorem 1: Suppose the $m\mathrm{D}(\boldsymbol{a}, f_i f_j, \varphi, d)$ prosection is performed, where $m \geq 2$ and $\varphi \in (0°, 90°)$. Then for any two vectors $\boldsymbol{f}^{\mathrm{A}} = (f_1^{\mathrm{A}}, \dots, f_m^{\mathrm{A}})$, and $\boldsymbol{f}^{\mathrm{B}} = (f_1^{\mathrm{B}}, \dots, f_m^{\mathrm{B}})$ inside the section the following holds. If

$$\boldsymbol{f}^{\mathrm{A}} \prec \boldsymbol{f}^{\mathrm{B}}$$

then

$$m\mathrm{D}(\boldsymbol{a}, f_i f_j, \varphi, d)(\boldsymbol{f}^{\mathrm{A}}) \prec m\mathrm{D}(\boldsymbol{a}, f_i f_j, \varphi, d)(\boldsymbol{f}^{\mathrm{B}}).$$

Proof: First, because $\boldsymbol{f}^{\mathrm{A}} \prec \boldsymbol{f}^{\mathrm{B}}$, it follows that $f_i^{\mathrm{A}} \leq f_i^{\mathrm{B}}$, $f_j^{\mathrm{A}} \leq f_j^{\mathrm{B}}$ and $f_{k_l}^{\mathrm{A}} \leq f_{k_l}^{\mathrm{B}}$ for $l \in \{1, \dots, m-2\}$. Also, because $\varphi \in (0°, 90°)$, $\sin\varphi > 0$ and $\cos\varphi > 0$. Therefore

$$s^{\mathrm{B}} - s^{\mathrm{A}} = \underbrace{(f_i^{\mathrm{B}} - f_i^{\mathrm{A}})}_{\geq 0}\underbrace{\cos\varphi}_{>0} + \underbrace{(f_j^{\mathrm{B}} - f_j^{\mathrm{A}})}_{\geq 0}\underbrace{\sin\varphi}_{>0} \geq 0.$$

This means that $\sigma^{\mathrm{A}} \preceq \sigma^{\mathrm{B}}$. Now, we only have to prove that $\sigma^{\mathrm{A}} \neq \sigma^{\mathrm{B}}$. If there exists an index l so that $f_{k_l}^{\mathrm{A}} < f_{k_l}^{\mathrm{B}}$, then $\sigma^{\mathrm{A}} \neq \sigma^{\mathrm{B}}$. Otherwise, if $f_{k_l}^{\mathrm{A}} = f_{k_l}^{\mathrm{B}}$ for all $l \in \{1, \dots, m-2\}$, then either $f_i^{\mathrm{A}} < f_i^{\mathrm{B}}$ or $f_j^{\mathrm{A}} < f_j^{\mathrm{B}}$. In either case this means that $s^{\mathrm{B}} - s^{\mathrm{A}} > 0$, which proves the theorem. ■

Theorem 2: Suppose the $m\mathrm{D}(\boldsymbol{a}, f_i f_j, \varphi, d)$ prosection is performed, where $m \geq 2$ and $\varphi \in (0°, 90°)$. Then for any two vectors $\boldsymbol{f}^{\mathrm{A}} = (f_1^{\mathrm{A}}, \dots, f_m^{\mathrm{A}})$, and $\boldsymbol{f}^{\mathrm{B}} = (f_1^{\mathrm{B}}, \dots, f_m^{\mathrm{B}})$ inside the section the following holds. If

$$m\mathrm{D}(\boldsymbol{a}, f_i f_j, \varphi, d)(\boldsymbol{f}^{\mathrm{A}}) \prec m\mathrm{D}(\boldsymbol{a}, f_i f_j, \varphi, d)(\boldsymbol{f}^{\mathrm{B}})$$

and

$$s(f_i^{\mathrm{B}}, f_j^{\mathrm{B}}) - s(f_i^{\mathrm{A}}, f_j^{\mathrm{A}}) \geq 2d \max\left\{\tan\varphi, \tan^{-1}\varphi\right\}$$

then

$$\boldsymbol{f}^{\mathrm{A}} \prec \boldsymbol{f}^{\mathrm{B}}.$$

Proof: First, let us show that under the assumptions of the theorem $\boldsymbol{f}^{\mathrm{A}} \preceq \boldsymbol{f}^{\mathrm{B}}$. Because $\sigma^{\mathrm{A}} \prec \sigma^{\mathrm{B}}$, it follows that $f_{k_l}^{\mathrm{A}} \leq f_{k_l}^{\mathrm{B}}$ for $l \in \{1, \dots, m-2\}$. We need to show that $f_i^{\mathrm{A}} \leq f_i^{\mathrm{B}}$ and $f_j^{\mathrm{A}} \leq f_j^{\mathrm{B}}$.

If $d > 0$, $(f_i^{\mathrm{A}}, f_j^{\mathrm{A}})$ is not the only vector to be projected into the value s^A. In fact, the whole line segment

$$f_j = a_j - \frac{f_i - a_i}{\tan\varphi} + \frac{s^A}{\sin\varphi}$$

where

$$\begin{aligned} f_i &\in [a_i + s^A\cos\varphi - d\sin\varphi, a_i + s^A\cos\varphi + d\sin\varphi] \\ f_j &\in [a_j + s^A\sin\varphi - d\cos\varphi, a_j + s^A\sin\varphi + d\cos\varphi] \end{aligned}$$

is projected into the same value s^A (see Fig. 22). We will denote this as line segment A. Analogously, the whole line segment

$$f_j = a_j - \frac{f_i - a_i}{\tan\varphi} + \frac{s^B}{\sin\varphi}$$

where

$$\begin{aligned} f_i &\in [a_i + s^B\cos\varphi - d\sin\varphi, a_i + s^B\cos\varphi + d\sin\varphi] \\ f_j &\in [a_j + s^B\sin\varphi - d\cos\varphi, a_j + s^B\sin\varphi + d\cos\varphi] \end{aligned}$$

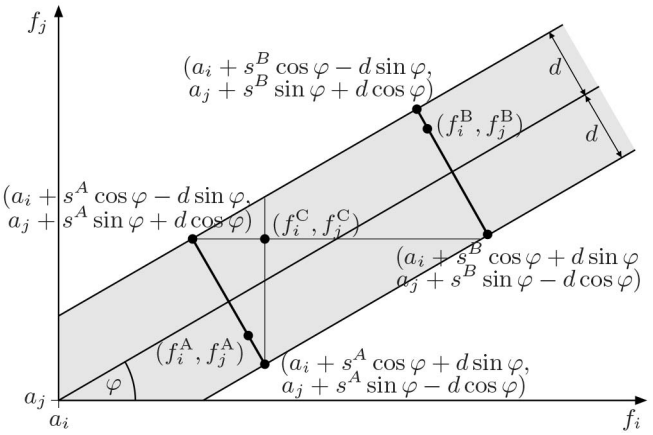

Fig. 22. Line segments A and B.

is projected into the value s^B. This is line segment B. Note that if $d = 0$, the line segments A and B consist of only one vector each, which are equal to $(a_i + s^A \cos\varphi, a_j + s^A \sin\varphi)$ and $(a_i + s^B \cos\varphi, a_j + s^B \sin\varphi)$, respectively.

Showing that all vectors from line segment A weakly dominate the whole line segment B proves that $f_i^{\mathrm{A}} \leq f_i^{\mathrm{B}}$ and $f_j^{\mathrm{A}} \leq f_j^{\mathrm{B}}$. Since the weak dominance relation is transitive, this can be done in two steps.

1) Line segment A weakly dominates a vector $(f_i^{\mathrm{C}}, f_j^{\mathrm{C}})$.
2) The vector $(f_i^{\mathrm{C}}, f_j^{\mathrm{C}})$ weakly dominates line segment B.

We can show that this holds for the vector

$$(f_i^{\mathrm{C}}, f_j^{\mathrm{C}}) = (a_i + s^A \cos\varphi + d\sin\varphi, a_j + s^A \sin\varphi + d\cos\varphi).$$

Proof of step 1: It is trivial to see that all vectors from the line segment A weakly dominate the vector $(f_i^{\mathrm{C}}, f_j^{\mathrm{C}})$.

Proof of step 2: The vector $(f_i^{\mathrm{C}}, f_j^{\mathrm{C}})$ weakly dominates the line segment B when the following two inequalities hold:

$$\begin{aligned} f_i^{\mathrm{C}} \leq a_i + s^B \cos\varphi - d\sin\varphi \quad f_j^{\mathrm{C}} &\leq a_j + s^B \sin\varphi - d\cos\varphi \\ f_i^{\mathrm{C}} &\leq a_i + s^B \cos\varphi - d\sin\varphi \\ a_i + s^A \cos\varphi + d\sin\varphi &\leq a_i + s^B \cos\varphi - d\sin\varphi \\ (s^B - s^A)\cos\varphi &\geq 2d\sin\varphi \\ s^B - s^A &\geq 2d\tan\varphi \end{aligned}$$

$$\begin{aligned} f_j^{\mathrm{C}} &\leq a_j + s^B \sin\varphi - d\cos\varphi \\ a_j + s^A \sin\varphi + d\cos\varphi &\leq a_j + s^B \sin\varphi - d\cos\varphi \\ (s^B - s^A)\sin\varphi &\geq 2d\cos\varphi \\ s^B - s^A &\geq 2d\tan^{-1}\varphi. \end{aligned}$$

Both inequalities hold because of the condition from the theorem

$$s^B - s^A \geq 2d\max\left\{\tan\varphi, \tan^{-1}\varphi\right\}.$$

Now, we only need to show that $\boldsymbol{f}^{\mathrm{A}} \neq \boldsymbol{f}^{\mathrm{B}}$. If there exists an index l so that $f_{k_l}^{\mathrm{A}} < f_{k_l}^{\mathrm{B}}$, then $\boldsymbol{f}^{\mathrm{A}} \neq \boldsymbol{f}^{\mathrm{B}}$. Otherwise, if $f_{k_l}^{\mathrm{A}} = f_{k_l}^{\mathrm{B}}$ for all $l \in \{1, \ldots, m-2\}$, then $s^{\mathrm{A}} < s^{\mathrm{B}}$. Since $\varphi \in (0°, 90°)$, $\sin\varphi > 0$ and $\cos\varphi > 0$

$$s^{\mathrm{B}} - s^{\mathrm{A}} = \underbrace{(f_i^{\mathrm{B}} - f_i^{\mathrm{A}})}_{\geq 0}\underbrace{\cos\varphi}_{>0} + \underbrace{(f_j^{\mathrm{B}} - f_j^{\mathrm{A}})}_{\geq 0}\underbrace{\sin\varphi}_{>0} > 0.$$

Because of this, $(f_i^{\mathrm{B}} - f_i^{\mathrm{A}})$ and $(f_j^{\mathrm{B}} - f_j^{\mathrm{A}})$ cannot be 0 at the same time, which means that $\boldsymbol{f}^{\mathrm{A}} \prec \boldsymbol{f}^{\mathrm{B}}$. ■

Theorem 3: Suppose the $m\mathrm{D}(\boldsymbol{a}, f_i f_j, \varphi, d)$ prosection is performed, where $m \geq 2$. Then for any two vectors $\boldsymbol{f}^{\mathrm{A}} = (f_1^{\mathrm{A}}, \ldots, f_m^{\mathrm{A}})$ and $\boldsymbol{f}^{\mathrm{B}} = (f_1^{\mathrm{B}}, \ldots, f_m^{\mathrm{B}})$ inside the section

$$\left\| m\mathrm{D}(\boldsymbol{a}, f_i f_j, \varphi, d)(\boldsymbol{f}^{\mathrm{A}}) - m\mathrm{D}(\boldsymbol{a}, f_i f_j, \varphi, d)(\boldsymbol{f}^{\mathrm{B}}) \right\| \leq \leq \left\| \boldsymbol{f}^{\mathrm{A}} - \boldsymbol{f}^{\mathrm{B}} \right\|.$$

The equality holds iff

$$\frac{f_j^{\mathrm{A}} - f_j^{\mathrm{B}}}{f_i^{\mathrm{A}} - f_i^{\mathrm{B}}} = \tan\varphi.$$

Proof: Let us first provide the proof for the inequality

$$\begin{aligned} &\left\| \boldsymbol{f}^{\mathrm{A}} - \boldsymbol{f}^{\mathrm{B}} \right\|^2 - \left\| \sigma^{\mathrm{A}} - \sigma^{\mathrm{B}} \right\|^2 = \\ &= (f_1^{\mathrm{A}} - f_1^{\mathrm{B}})^2 + \cdots + (f_m^{\mathrm{A}} - f_m^{\mathrm{B}})^2 - \\ &- \left((s^{\mathrm{A}} - s^{\mathrm{B}})^2 + (f_{k_1}^{\mathrm{A}} - f_{k_1}^{\mathrm{B}})^2 + \cdots + (f_{k_{m-2}}^{\mathrm{A}} - f_{k_{m-2}}^{\mathrm{B}})^2 \right) = \\ &= (f_i^{\mathrm{A}} - f_i^{\mathrm{B}})^2 + (f_j^{\mathrm{A}} - f_j^{\mathrm{B}})^2 - (s^{\mathrm{A}} - s^{\mathrm{B}})^2 = \\ &= (f_i^{\mathrm{A}} - f_i^{\mathrm{B}})^2 + (f_j^{\mathrm{A}} - f_j^{\mathrm{B}})^2 - \\ &- \left((f_i^{\mathrm{A}} - f_i^{\mathrm{B}})\cos\varphi + (f_j^{\mathrm{A}} - f_j^{\mathrm{B}})\sin\varphi \right)^2 = \\ &= \left((f_i^{\mathrm{A}} - f_i^{\mathrm{B}})^2 + (f_j^{\mathrm{A}} - f_j^{\mathrm{B}})^2 \right)(\sin^2\varphi + \cos^2\varphi) - \\ &- \left((f_i^{\mathrm{A}} - f_i^{\mathrm{B}})\cos\varphi + (f_j^{\mathrm{A}} - f_j^{\mathrm{B}})\sin\varphi \right)^2 = \\ &= (f_i^{\mathrm{A}} - f_i^{\mathrm{B}})^2 \sin^2\varphi + (f_j^{\mathrm{A}} - f_j^{\mathrm{B}})^2 \sin^2\varphi + \\ &+ (f_i^{\mathrm{A}} - f_i^{\mathrm{B}})^2 \cos^2\varphi + (f_j^{\mathrm{A}} - f_j^{\mathrm{B}})^2 \cos^2\varphi - \\ &- (f_i^{\mathrm{A}} - f_i^{\mathrm{B}})^2 \cos^2\varphi - 2(f_i^{\mathrm{A}} - f_i^{\mathrm{B}})(f_j^{\mathrm{A}} - f_j^{\mathrm{B}})\sin\varphi\cos\varphi - \\ &- (f_j^{\mathrm{A}} - f_j^{\mathrm{B}})^2 \sin^2\varphi = \\ &= (f_i^{\mathrm{A}} - f_i^{\mathrm{B}})^2 \sin^2\varphi + (f_j^{\mathrm{A}} - f_j^{\mathrm{B}})^2 \cos^2\varphi - \\ &- 2(f_i^{\mathrm{A}} - f_i^{\mathrm{B}})(f_j^{\mathrm{A}} - f_j^{\mathrm{B}})\sin\varphi\cos\varphi = \\ &= \left((f_i^{\mathrm{A}} - f_i^{\mathrm{B}})\sin\varphi - (f_j^{\mathrm{A}} - f_j^{\mathrm{B}})\cos\varphi \right)^2 \geq 0. \end{aligned}$$

This means that $\left\| \boldsymbol{f}^{\mathrm{A}} - \boldsymbol{f}^{\mathrm{B}} \right\|^2 \geq \left\| \sigma^{\mathrm{A}} - \sigma^{\mathrm{B}} \right\|^2$, from which it follows that $\left\| \boldsymbol{f}^{\mathrm{A}} - \boldsymbol{f}^{\mathrm{B}} \right\| \geq \left\| \sigma^{\mathrm{A}} - \sigma^{\mathrm{B}} \right\|$.

The equality holds when

$$(f_i^{\mathrm{A}} - f_i^{\mathrm{B}})\sin\varphi - (f_j^{\mathrm{A}} - f_j^{\mathrm{B}})\cos\varphi = 0$$

which is equivalent to

$$\frac{f_j^{\mathrm{A}} - f_j^{\mathrm{B}}}{f_i^{\mathrm{A}} - f_i^{\mathrm{B}}} = \tan\varphi.$$

■

Theorem 4: Suppose the $m\mathrm{D}(\boldsymbol{a}, f_i f_j, \varphi, d)$ prosection is performed, where $m \geq 2$. Let $\boldsymbol{f}^{\mathrm{A}} = (f_1^{\mathrm{A}}, \ldots, f_m^{\mathrm{A}})$, $\boldsymbol{f}^{\mathrm{B}} = (f_1^{\mathrm{B}}, \ldots, f_m^{\mathrm{B}})$ and $\boldsymbol{f}^{\mathrm{R}} = (f_1^{\mathrm{R}}, \ldots, f_m^{\mathrm{R}})$ be three vectors inside

the section and let us assume that $\left\|\boldsymbol{f}^{\mathrm{A}}-\boldsymbol{f}^{\mathrm{R}}\right\| < \left\|\boldsymbol{f}^{\mathrm{B}}-\boldsymbol{f}^{\mathrm{R}}\right\|$. If $\left\|\boldsymbol{f}^{\mathrm{B}}-\boldsymbol{f}^{\mathrm{R}}\right\|^2 - \left\|\boldsymbol{f}^{\mathrm{A}}-\boldsymbol{f}^{\mathrm{R}}\right\|^2 > 4D^2$, then

$$\left\|m\mathrm{D}(\boldsymbol{a},f_if_j,\varphi,d)(\boldsymbol{f}^{\mathrm{A}}) - m\mathrm{D}(\boldsymbol{a},f_if_j,\varphi,d)(\boldsymbol{f}^{\mathrm{R}})\right\| < \left\|m\mathrm{D}(\boldsymbol{a},f_if_j,\varphi,d)(\boldsymbol{f}^{\mathrm{B}}) - m\mathrm{D}(\boldsymbol{a},f_if_j,\varphi,d)(\boldsymbol{f}^{\mathrm{R}})\right\|.$$

On the other hand, if

$$\left\|m\mathrm{D}(\boldsymbol{a},f_if_j,\varphi,d)(\boldsymbol{f}^{\mathrm{B}}) - m\mathrm{D}(\boldsymbol{a},f_if_j,\varphi,d)(\boldsymbol{f}^{\mathrm{R}})\right\|^2 - - \left\|m\mathrm{D}(\boldsymbol{a},f_if_j,\varphi,d)(\boldsymbol{f}^{\mathrm{A}}) - m\mathrm{D}(\boldsymbol{a},f_if_j,\varphi,d)(\boldsymbol{f}^{\mathrm{R}})\right\|^2 > 4d^2$$

then $\left\|\boldsymbol{f}^{\mathrm{A}}-\boldsymbol{f}^{\mathrm{R}}\right\| < \left\|\boldsymbol{f}^{\mathrm{B}}-\boldsymbol{f}^{\mathrm{R}}\right\|$.

Proof: First, note that for any vector in the section the following holds:

$$\left\|\sigma-\sigma^{\mathrm{R}}\right\|^2 + 4d^2 \geq \left\|\boldsymbol{f}-\boldsymbol{f}^{\mathrm{R}}\right\|^2 \geq \left\|\sigma-\sigma^{\mathrm{R}}\right\|^2.$$

Let us provide the proof for the first affirmation. From the previous inequality and the assumption in the theorem it follows that:

$$\left\|\sigma^{\mathrm{B}}-\sigma^{\mathrm{R}}\right\|^2 + 4d^2 \geq \left\|\boldsymbol{f}^{\mathrm{B}}-\boldsymbol{f}^{\mathrm{R}}\right\|^2 > \left\|\boldsymbol{f}^{\mathrm{A}}-\boldsymbol{f}^{\mathrm{R}}\right\|^2 + 4d^2.$$

Therefore, $\left\|\sigma^{\mathrm{B}}-\sigma^{\mathrm{R}}\right\| > \left\|\boldsymbol{f}^{\mathrm{A}}-\boldsymbol{f}^{\mathrm{R}}\right\|$.

The second affirmation can be proven in a similar way using the same inequality as before and the assumption in the theorem

$$\left\|\boldsymbol{f}^{\mathrm{B}}-\boldsymbol{f}^{\mathrm{R}}\right\|^2 \geq \left\|\sigma^{\mathrm{B}}-\sigma^{\mathrm{R}}\right\|^2 > \left\|\sigma^{\mathrm{A}}-\sigma^{\mathrm{R}}\right\|^2 + 4d^2 \geq \geq \left\|\boldsymbol{f}^{\mathrm{A}}-\boldsymbol{f}^{\mathrm{R}}\right\|^2.$$

This means that $\left\|\boldsymbol{f}^{\mathrm{B}}-\boldsymbol{f}^{\mathrm{R}}\right\| > \left\|\boldsymbol{f}^{\mathrm{A}}-\boldsymbol{f}^{\mathrm{R}}\right\|$. ■

ACKNOWLEDGMENT

T. Tušar would like to thank B. Naujoks for providing comments on the draft version of the manuscript.

REFERENCES

[1] K. Deb and S. Jain, "Running performance metrics for evolutionary multi-objective optimizations," in *Proc. 4th Asia-Pacific Conf. SEAL*, 2002, pp. 13–20.

[2] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. D. Grunert da Fonseca, "Performance assessment of multiobjective optimizers: An analysis and review," *IEEE Trans. Evol. Comput.*, vol. 7, no. 2, pp. 117–132, Apr. 2003.

[3] J. Knowles, D. Corne, and K. Deb, "Introduction: Problem solving, EC and EMO," in *Multiobjective Problem Solving from Nature: From Concepts to Applications*, J. Knowles, D. Corne, and K. Deb, Eds. Berlin, Germany: Springer, 2008, pp. 1–28.

[4] W. Metzger, *Laws of Seeing*. Cambridge, MA, USA: MIT Press, 2006.

[5] M. Köppen and K. Yoshida, "Visualization of Pareto-sets in evolutionary multi-objective optimization," in *Proc. 7th Int. Conf. HIS*, Kaiserslautern, Germany, 2007, pp. 156–161.

[6] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scalable test problems for evolutionary multi-objective optimization," in *Evolutionary Multiobjective Optimization: Theoretical Advances and Applications*, A. Abraham, R. Jain, and R. Goldberg, Eds. Berlin, Germany: Springer, 2005, pp. 105–145.

[7] S. Huband, P. Hingston, L. Barone, and L. While, "A review of multiobjective test problems and a scalable test problem toolkit," *IEEE Trans. Evol. Comput.*, vol. 10, no. 5, pp. 477–506, Oct. 2006.

[8] R. Y. Rubinstein and D. P. Kroese, *Simulation and the Monte Carlo Method.* Hoboken, NJ, USA: Wiley, 2008.

[9] S. R. Dos Santos, "A framework for the visualization of multidimensional and multivariate data," Ph.D. dissertation, School Comput., Univ. Leeds, Leeds, U.K., 2004.

[10] K. Miettinen, "Survey of methods to visualize alternatives in multiple criteria decision making problems," *OR Spectr.*, vol. 36, no. 1, pp. 3–37, 2014.

[11] M. F. Ashby, "Multi-objective optimization in material design and selection," *Acta Materialia*, vol. 48, no. 1, pp. 359–369, 2000.

[12] S. Poles, P. Geremia, F. Campos, S. Weston, and M. Islam, "MOGA-II for an automotive cooling duct optimization on distributed resources," in *Proc. 4th Int. Conf. EMO*, Matsushima, Japan, 2007, pp. 633–644.

[13] M. Liebscher, K. Witowski, and T. Goel, "Decision making in multi-objective optimization for industrial applications—Data mining and visualization of Pareto data," in *Proc. 8th World Congr. SMO*, Lisbon, Portugal, 2009.

[14] P. E. Hoffman, G. G. Grinstein, K. Marx, I. Grosse, and E. Stanley, "DNA visual and analytic data mining," in *Proc. IEEE Vis. Conf.*, Phoenix, AZ, USA, 1997, pp. 437–441.

[15] D. J. Walker, J. E. Fieldsend, and R. M. Everson, "Visualising many-objective populations," in *Proc. 14th Int. Conf. GECCO*, Philadelphia, PA, USA, 2012, pp. 451–458.

[16] D. J. Walker, R. M. Everson, and J. E. Fieldsend, "Visualizing mutually nondominating solution sets in many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 17, no. 2, pp. 165–184, Apr. 2013.

[17] A. Inselberg, *Parallel Coordinates: Visual Multidimensional Geometry and its Applications*. New York, NY, USA: Springer, 2009.

[18] A. Pryke, S. Mostaghim, and A. Nazemi, "Heatmap visualisation of population based multi objective algorithms," in *Proc. 4th Int. Conf. EMO*, Matsushima, Japan, 2007, pp. 361–375.

[19] J. W. Sammon, "A non-linear mapping for data structure analysis," *IEEE Trans. Comput.*, vol. C-18, no. 5, pp. 401–408, May 1969.

[20] J. Valdés and A. Barton, "Visualizing high dimensional objective spaces for multi-objective optimization: A virtual reality approach," in *Proc. CEC*, Singapore, 2007, pp. 4199–4206.

[21] D. Lowe and M. E. Tipping, "Feed-forward neural networks and topographic mappings for exploratory data analysis," *Neural Comput. Appl.*, vol. 4, no. 2, pp. 83–95, 1996.

[22] R. M. Everson and J. E. Fieldsend, "Multi-class ROC analysis from a multi-objective optimisation perspective," *Pattern Recognit. Lett.*, vol. 27, no. 8, pp. 918–927, 2006.

[23] T. Kohonen, *Self-Organizing Maps.* Berlin, Germany: Springer, 2001.

[24] S. Obayashi and D. Sasaki, "Visualization and data mining of Pareto solutions using self-organizing map," in *Proc. 2nd Int. Conf. EMO*, Faro, Portugal, 2003, pp. 796–809.

[25] M. Yamamoto, T. Yoshikawa, and T. Furuhashi, "Study on effect of MOGA with interactive island model using visualization," in *Proc. CEC*, Barcelona, Spain, 2010, pp. 1–6.

[26] J. B. Tenenbaum, V. Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[27] I. Borg and P. J. F. Groenen, *Modern Multidimensional Scaling: Theory and Applications.* New York, NY, USA: Springer, 2005.

[28] F. Kudo and T. Yoshikawa, "Knowledge extraction in multi-objective optimization problem based on visualization of Pareto solutions," in *Proc. CEC*, Brisbane, QLD, Australia, 2012, pp. 1–6.

[29] K. H. Ang, G. Chong, and Y. Li, "Visualization technique for analyzing non-dominated set comparison," in *Proc. 4th Asia-Pacific Conf. SEAL*, 2002, pp. 36–40.

[30] A. V. Lotov, V. A. Bushenkov, and G. K. Kamenev, *Interactive Decision Maps: Approximation and Visualization of Pareto Frontier.* New York, NY, USA: Springer, 2004.

[31] A. V. Lotov and K. Miettinen, "Visualizing the Pareto frontier," in *Multiobjective Optimization.* Berlin, Germany: Springer, 2008, pp. 213–243.

[32] C. M. Fonseca and P. J. Fleming, "On the performance assessment and comparison of stochastic multiobjective optimizers," in *Proc. 4th Int. Conf. PPSN*, Berlin, Germany, 1996, pp. 584–593.

[33] J. Knowles, "A summary-attainment-surface plotting method for visualizing the performance of stochastic multiobjective optimizers," in *Proc. 5th Int. Conf. ISDA*, Washington, DC, USA, 2005, pp. 552–557.

[34] G. Agrawal *et al.*, "Intuitive visualization of Pareto frontier for multi-objective optimization in n-dimensional performance space," in *Proc. 10th AIAA/ISSMO*, 2004.

[35] G. Agrawal, C. L. Bloebaum, and K. Lewis, "Intuitive design selection using visualized n-dimensional Pareto frontier," in *Proc. 46th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, Materials Conf.*, 2005, pp. 1–14.

[36] K. Deb, A. Pratap, S. Agrawal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA–II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.
[37] X. Blasco, J. M. Herrero, J. Sanchis, and M. Martínez, "A new graphical visualization of n-dimensional Pareto front for decision-making in multiobjective optimization," *Inf. Sci.*, vol. 178, no. 20, pp. 3908–3924, Oct. 2008.
[38] P.-W. Chiu and C. Bloebaum, "Hyper-radial visualization (HRV) method with range-based preferences for multi-objective decision making," *Struct. Multidiscip. Optim.*, vol. 40, nos. 1–6, pp. 97–115, 2010.
[39] D. J. Walker, R. M. Everson, and J. E. Fieldsend, "Visualisation and ordering of many-objective populations," in *Proc. CEC*, Barcelona, Spain, 2010, pp. 1–8.
[40] L. Tweedie, R. Spence, H. Dawkes, and H. Su, "Externalising abstract mathematical models," in *Proc. Conf. SIGCHI*, San Francisco, CA, USA, 1996, pp. 406–412.
[41] G. W. Furnas and A. Buja, "Prosection views: Dimensional inference through sections and projections," *J. Comput. Graph. Stat.*, vol. 3, no. 4, pp. 323–353, 1994.
[42] T. Tušar and B. Filipič, "Visualizing 4D approximation sets of multiobjective optimizers with prosections," in *Proc. 13th Ann. GECCO*, 2011, pp. 737–744.
[43] K. Deb and D. K. Saxena, "Searching for Pareto-optimal solutions through dimensionality reduction for certain large-dimensional multiobjective optimization problems," in *Proc. CEC*, 2006, pp. 3352–3360.
[44] D. K. Saxena, J. A. Duro, A. Tiwari, K. Deb, and Q. Zhang, "Objective reduction in many-objective optimization: Linear and nonlinear algorithms," *IEEE Trans. Evol. Comput.*, vol. 17, no. 1, pp. 77–99, Feb. 2013.
[45] J. Branke, K. Deb, H. Dierolf, and M. Oswald, "Finding knees in multiobjective optimization," in *Proc. 8th Int. Conf. PPSN*, Birmingham, U.K., 2004, pp. 722–731.
[46] T. Robič and B. Filipič, "DEMO: Differential evolution for multiobjective optimization," in *Proc. 3rd Int. Conf. EMO*, Guanajuato, Mexico, 2005, pp. 520–533.

Tea Tušar (S'12) received the B.Sc. degree in applied mathematics and the M.Sc. degree in computer and information science, both from University of Ljubljana, Ljubljana, Slovenia. She is currently working toward the Ph.D. degree in information and communication technologies from Jožef Stefan International Postgraduate School, Ljubljana.

She is a Research Assistant with the Department of Intelligent Systems, Jožef Stefan Institute, Ljubljana. Her research interests include evolutionary algorithms for single- and multiobjective optimization with applications in optimization of production processes and engineering design, and also visualization techniques used in multiobjective optimization.

Bogdan Filipič (M'93) received the B.Sc., M.Sc., and Ph.D. degrees in computer and information science from University of Ljubljana, Ljubljana, Slovenia, in 1983, 1989, and 1993, respectively.

Since 2002, he has been a Senior Research Associate and, since 2011, he was the Head of the Computational Intelligence Group, Department of Intelligent Systems, Jožef Stefan Institute, Ljubljana, Slovenia. He is also an Associate Professor of Computer and Information Science with teaching experience with University of Ljubljana, University of Nova Gorica, Jožef Stefan International Postgraduate School, Ljubljana, and University of Oulu, Oulu, Finland. His research interests include stochastic optimization, evolutionary computation, and intelligent data analysis.