IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

*Digital Object Identifier 10.1109/ACCESS.2024.0429000*

# SCORE Band Visualizations: Supporting Decision Makers in Comparing High-Dimensional Outcome Vectors in Multiobjective Optimization

**BHUPINDER S. SAINI[1], KAISA MIETTINEN[1], KATHRIN KLAMROTH[2], RALPH E. STEUER[3], KERSTIN DÄCHERT[4]**

[1]University of Jyvaskyla, Faculty of Information Technology, P.O. Box 35 (Agora), FI-40014 University of Jyvaskyla, Finland
[2]University of Wuppertal, School of Mathematics and Natural Sciences, Gaußstr. 20, 42119 Wuppertal, Germany
[3]Department of Finance, University of Georgia, Athens GA 30602, USA
[4]Hochschule für Technik und Wirtschaft Dresden - University of Applied Sciences, Friedrich-List-Platz 1, 01069 Dresden, Germany

Corresponding author: Bhupinder S. Saini (e-mail: bhupinder.s.saini@jyu.fi).

This research was partly funded by the Research Council of Finland under the grants 322221 and 355346.

**ABSTRACT** Clearly arranged visualizations are needed in multiobjective optimization problems with a large number of objective functions, when a large number of Pareto optimal outcome vectors (vectors of objective function values) must be compared during the decision making processes. This paper contributes to visualizing such outcome vectors independent of how they have been generated.
Parallel coordinate plots are a widely used visualization technique to represent different outcome vectors. We propose a novel visualization technique called SCORE bands to be used with parallel coordinate plots to support the decision maker in simultaneously identifying patterns in outcome vectors and correlations among the objective functions in a meaningful way. To do so, amongst others, we change the ordering of objective functions and modify the distances among them in parallel coordinate plots. SCORE bands also have interactive capabilities allowing the decision maker to first study general trends among the outcome vectors as bands and then zoom-in and move about different groups of outcome vectors of interest. The novelty of our approach lies in proposing a visually appealing way to support the decision maker in dealing with large amounts of information. We demonstrate the benefits of SCORE bands with different examples.

**INDEX TERMS** Multiple criteria optimization, Interactive visualization, Correlated objectives, Parallel coordinate plots, Pareto optimality.

## I. INTRODUCTION

The aim of multiobjective optimization methods is to support a domain expert, to be referred to as a decision maker (DM), in finding the best balance among conflicting objective functions. Many methods generate so-called Pareto optimal solutions and corresponding outcome vectors (i.e., vectors of objective function values), where no objective function can be improved without impairing at least one of the others. Different methods generate varying amounts of Pareto optimal solutions. Typically, the corresponding outcome vectors are to be compared by a DM. The task of comparison gets more demanding when the numbers of objective functions and outcome vectors increase.

Carefully selected visualizations can help a DM in gaining insight in different trade-offs among outcome vectors. Means of visualization for multiobjective optimization purposes are surveyed, e.g., in Gettinger, Kiesling, Stummer, *et al.* [13], Korhonen and Wallenius [24], Lotov and Miettinen [29], Miettinen [33], and Woodruff, Reed, and Simpson [50]. Examples of popular visualization techniques are parallel coordinate plots also known as value paths in Cohon [7] and Heinrich and Weiskopf [17], spider web charts, scatterplot matrices, petal diagrams, star coordinate plots and glyphs. Further techniques include heatmaps [19], knowCube [44], interactive decision maps [28], the prosection method [45], PaletteViZ [43], 3D-RadVis [22], and barycentric RadViz [47]. There are also, e.g., Andrews plots [1] and Chernoff faces [5] from the early visualization days of multivariate data but they are, unfortunately, of little help in the display of trade-off information which is the focus of this paper.

As discussed e.g., in Fonseca, Antunes, Lacour, *et al.* [12], visualizations can be applied for various purposes in multiobjective optimization ranging from following the progress of the solution process to visualizing uncertainty and to identifying information to be visualized. Here we focus on visualizing Pareto optimal outcome vectors. Recent developments in visualization methods include open-source building blocks for implementing parallel coordinate plots [38] and the needs encountered in specific application domains, e.g., [3], [6], [11], [14], [27], [32]. In many studies, parallel coordinate plots have been found useful in visualizing outcome vectors supported by, e.g, clustering [4], [51]. In parallel coordinate plots, objective functions are typically represented by vertical axes and outcome vectors are represented by polylines. Naturally, the order of the axes affects the interpretability since trade-offs in the objective functions that are next to each other are easier to inspect. Even though Ankerst, Berchtold, and Keim [2] is not about multiobjective optimization, the measures of similarity proposed could be used to order objective functions in a parallel coordinate plot. An approach for deriving the order of the axes using Spearman's rank correlation was proposed in Zhen, Li, Cheng, *et al.* [52]. Another approach is presented in Huang and Siraj [21], which orders the axes to minimize the number of "cross-overs" of polylines of different outcome vectors, thus minimizing the complexity of the visualization. In both Ankerst, Berchtold, and Keim [2] and Zhen, Li, Cheng, *et al.* [52], the axes are equidistant, as they are also in Huang and Siraj [21] and Smedberg and Bandaru [42]. Additionally, the approach of Huang and Siraj [21] is only applicable to visualize datasets with a small number of outcome vectors due to expensive underlying calculations.

In this paper, we propose means for supporting a DM in understanding the information contained in any collection of Pareto optimal outcome vectors. Our aim is to show both correlation among objective functions and clusters in outcome vectors so that it is easier to digest major insights. Roughly speaking, we visually cluster both objective functions and outcome vectors. We propose to use Pearson correlation coefficients of all objective function pairs to calculate the order of the objective functions. Moreover, we visualize the correlation information by changing the distance between neighboring axes based upon the value of the Pearson correlation coefficients of the corresponding objective functions. Thus, we not only determine the order in which the objective functions should be displayed but illustrate correlation information visually. This is particularly helpful when the number of objective functions is above three. While the ideas behind this research are reported in Dächert, Klamroth, Miettinen, *et al.* [9] (applying different clustering tools), this is the full development of the initiatives outlined in that piece (including the idea about modifying the distances between axes in a parallel coordinate plot, which does not appear elsewhere other than in Dächert, Klamroth, Miettinen, *et al.* [9]).

When the number of outcome vectors is high, one can filter out undesired ones as, e.g., in knowCube [44]. But if one wants to understand better what kinds of trade-offs are represented in the data available, as we do in this paper, clustering can be applied to first show the bigger trends among the objective functions [31], [35], [53]. Then the DM can be given the ability to zoom in on clusters of special interest to examine individual outcome vectorsmore closely. In support of this kind of analysis, we propose the use of SCORE band visualizations.

Overall, our novel contribution to visualize Pareto optimal outcome vectors with modified parallel coordinate plots is three-fold: (i) to order the objective functions (to visually group similar objective functions in a manner beneficial to a DM), (ii) to reflect correlation among the objective functions by varying the distances between the axes in the visualization, and (iii) to apply SCORE bands to visualize outcome vectors in a visually pleasing way by showing major trends. SCORE bands differs from other state-of-the-art methods in various ways. Unlike most visualizations that are based on the parallel coordinate plot, SCORE bands has an explicit focus on decision making, requiring us to both utilize existing ideas as well as come up with new innovations to support the goal of decision-making. Secondly, unlike most visualizations aimed at aiding decision making, SCORE bands scales well to a large number of outcome vectors, enabling its usage with, for example, evolutionary approaches, which generate a large number of outcome vectors. Finally, unlike most previously mentioned visualizations, we have implemented the new visualization technique in an open-source Python package and make it available freely via the DESDEO framework [34] (https://desdeo.it.jyu.fi). This enables researchers and DMs to utilize SCORE bands with ease.

When especially compared against parallel coordinate plots, SCORE band visualization adopts a so-called "maximum effect for minimum means" approach [8], [18]. According to Ware [48], "the ultimate goal of interactive visualization design is to optimize applications so that they help us perform cognitive work more efficiently." We have carefully considered the costs and benefits of utilizing SCORE bands as a decision-making tool, as compared to parallel coordinate plots. We provide great benefits to the DM by providing correlation and clustering information. We keep the costs low by providing the information in an easy-to-digest manner, saving the DM's time. SCORE bands visualization minimizes the cognitive load on the DM while supporting the DM in analyzing the data.

We do not care how the set of outcome vectors to be visualized has been generated (e.g., by using scalarization-based or evolutionary approaches) as long as the vectors do not dominate one another[1]. The new visualizations can support the DM during a solution process to get an overall understanding or focus on outcome vectors of interest, provide preferences as well as identify a final, most preferred outcome vector. Thus, the proposed visualizations can be used with different multiobjective optimization methods.

The rest of this paper is structured as follows. In Section 2, we introduce main concepts and notation, and give an overview of related literature on parallel coordinate plots. We

[1] In general, the data to be visualized do not need to specifically come from a multiobjective optimization problem. The data can come from any multiple criteria decision making problem.

propose our new ways of visualizing sets of Pareto optimal outcome vectors, culminating in SCORE bands, in Section 3. We give examples in Section 4 as well as discuss different way of utilizing the new visualizations. Finally, we conclude in Section 5.

## II. PROBLEM FORMULATION AND RELATED LITERATURE

We consider *multiobjective optimization problems*

$$\min\{\boldsymbol{f}(\boldsymbol{x}) = (f_1(\boldsymbol{x}), ..., f_k(\boldsymbol{x})) \,:\, \boldsymbol{x} \in X\} \qquad \text{(MOP)}$$

with $k \geq 2$ real-valued objective functions $f_i : X \to \mathbb{R}$, $i = 1, \ldots, k$ and a feasible set $X$. The objective functions may represent, for example, economical and ecological goals, or the cost and the quality of a decision vector. Using the fact that maximization problems can be reformulated as equivalent minimization problems, we assume throughout this paper that all objective functions are to be minimized. Accordingly, we assume that a rational DM prefers smaller objective function values over larger objective function values in all objective functions. This is reflected in the concept of *Pareto optimality* .Pareto optimality can be best detected by comparing *outcome vectors* of feasible decision vectors in the $k$-dimensional *objective space* $\mathbb{R}^k$: An outcome vector $\boldsymbol{z}^1 = \boldsymbol{f}(\boldsymbol{x}^1)$ *dominates* $\boldsymbol{z}^2 = \boldsymbol{f}(\boldsymbol{x}^2)$ in $\mathbb{R}^k$ if and only if $z_i^1 \leq z_i^2$ for all $i = 1, \ldots, k$ and $\boldsymbol{z}^1 \neq \boldsymbol{z}^2$. Then a feasible decision vector $\boldsymbol{x} \in X$ is a Pareto optimal decision vector (not to be confused with its image, a Pareto optimal outcome vector) if and only if there is no other feasible decision vector $\bar{\boldsymbol{x}} \in X$ such that $\boldsymbol{f}(\bar{\boldsymbol{x}})$ dominates $\boldsymbol{f}(\boldsymbol{x})$. Corresponding to the set of all Pareto optimal decision vectors, i.e., the *Pareto set*, we have their respective images in the objective space constituting a *Pareto front*, i.e., the set of all Pareto optimal outcome vectors of problem (MOP).

To support the DM in the selection of the most preferred outcome vector, a clearly arranged graphic presentation of a representative subset of the Pareto front is crucial. This is particularly true when the number of objective functions increases, i.e., when $k$ is (considerably) larger than 2 or 3. Indeed, while for biobjective problems the Pareto front can be visualized in a 2-dimensional scatter plot that immediately shows the *trade-offs* between the two objective functions, this is no longer true for higher-dimensional problems. Already in a three-objective case, a direct visualization of the Pareto front in the objective space is difficult[2], and it is generally not useful at all in the case of more than 3 objective functions. At the same time, optimization problems with an increasing number of objective functions are becoming more and more relevant and popular in practical applications, and hence there is a growing need for efficient and meaningful presentations of Pareto optimal outcome vectors.

We focus on visualizations of a finite set of Pareto optimal outcome vectors or outcome vectors that do not dominate each other (i.e., nondominated vectors) on parallel coordinate plots (see, e.g., Cohon [7], Inselberg and Dimsdale [23], and Wegman [49]). As mentioned, each objective function is associated with a (vertical) axis that represents the range of possible function values, while each outcome vector is represented by its values on these axes and connected by a polyline known as a *value path*. Early implementations of parallel coordinate plots (see, e.g., Korhonen and Laakso [25], [26]) have generally ordered the axes in the plot in the same order as the objective functions in the problem definition. See Figure 1 for an illustration.

In the following, we provide a brief review on the related literature on parallel coordinate plots. Interpreting parallel coordinate plots visualizing a large number of (nondominated) outcome vectors can be challenging due to multiple issues. The first issue is the visualization of correlation information. The interpretation of correlation is paramount in decision making in multiobjective optimization problems as correlations represent trade-offs among the objective functions. In parallel coordinate plots, correlations are not observed directly. Instead, the information is presented implicitly via the value paths. If two highly correlated objective functions are placed next to each other in a plot, the value paths between the corresponding axes have a parallel relationship to each other. Instead, if the objective functions are negatively correlated, the value paths intersect between the two axes. While the correlation information about neighboring objective functions can be implicitly judged, correlation information about non-neighboring objective functions is lost in the visualization. Therefore, the order of objective functions in the plot plays a vital role in helping a DM interpret the insight within the outcome vectors. An approach for deriving the order of the axes using Spearman's rank correlation was proposed in Zhen, Li, Cheng, *et al.* [52].

Another issue is that of the overcrowding of value paths. Visualizing many value paths in the same space makes it difficult to follow any single value path. One way to solve this problem is to identify clusters of outcome vectors. The clustering information can then be visualized in the parallel coordinate plot to make interpretation easier. The simplest way to present this information is to color value paths belonging to different clusters differently. While this technique makes identifying clusters of outcome vectors easier, it does not reduce the visual clutter that arises due to the many value paths.

Another way to reduce visual clutter is to change the way outcome vectors are visualized. Parallel coordinate plots with bundling [31], [54] are a prime example. Instead of representing outcome vectors as polylines, these visualization techniques bundle together value paths that belong to the same cluster. This is achieved by curving the value paths inwards (towards the value path of the cluster centroid) between each pair of neighboring objective axes. Such bundles are easy to distinguish, making the interpretation of clusters easier. However, bundling can make the interpretation of correlation information, represented by the amount of crossing over of the value paths, more challenging [16]. McDonnell and

[2]Visualization in the form of 3-dimensional scatter plots can be useful to the DM if they can interact with (rotate, pan, and zoom) the plot. A non-interactive, i.e., static version of the same visualization can be challenging to interpret.

Mueller [31] present an alternative way to represent clusters in parallel coordinate plots. Instead of visualizing individual outcome vectors, they visualize the whole cluster as a semi-transparent band, where the thickness of the band represents the standard deviation of the outcome vector belonging to the cluster. However, this visualization technique also hides correlation information, making decision making challenging.

In the following section, we introduce SCORE bands and show how they address the challenges mentioned above.

## III. SCORE BANDS: SIMULTANEOUS CLUSTERING AND CORRELATED OBJECTIVE VISUALIZATION VIA BANDS

In this section, we describe the algorithm by which SCORE bands and their visualizations are constructed so as to enable the re-imagination of the concept of a parallel coordinate plot of this paper. The purpose is to support decision making in multiobjective optimization with visualizations of sets of Pareto optimal (or nondominated) outcome vectors in order to highlight key information contained in them while minimizing clutter that might otherwise distract from the decision making process. Bear in mind that while the goal of a visualization is to tailor it to the needs of a given DM, the algorithm may have to process many Pareto optimal outcome vectors. Possessing parameters designed to be easy to operate (to adapt to the needs of the problem at hand and the DM by e.g. an analyst supporting a DM), the algorithm consists of four steps:

1) **Outcome vector clustering**: Find clusters in the outcome vectors. This information helps the DM understand the distribution of outcome vectors in the objective space.
2) **Axis ordering**: Calculate an optimal ordering for the objective function axes of the parallel coordinate plot visualization. This helps visualizing information about the relationships between different objective functions, such as trade-offs or correlations.
3) **Inter-axis spacing**: Expand or contract the space between the objective function axes to highlight or suppress the relationship between neighboring objective functions in the parallel coordinate plot. This usage of the inter-axes space can help the DM focus on relationships among the outcome vectors or objective functions that they find important.
4) **Visualization via bands**: Visualize the information extracted in the previous three steps in an effective, visually accessible, and pleasing manner. Instead of visualizing individual outcome vectors, only visualize the clusters of outcome vectors using bands by default. Enable the DM to show/hide the bands or value paths of individual outcome vectors in an interactive manner.

Other simple visualizations can be presented to a DM along with SCORE bands. They can provide additional insight into the problem and make it easier to understand SCORE band visualizations.

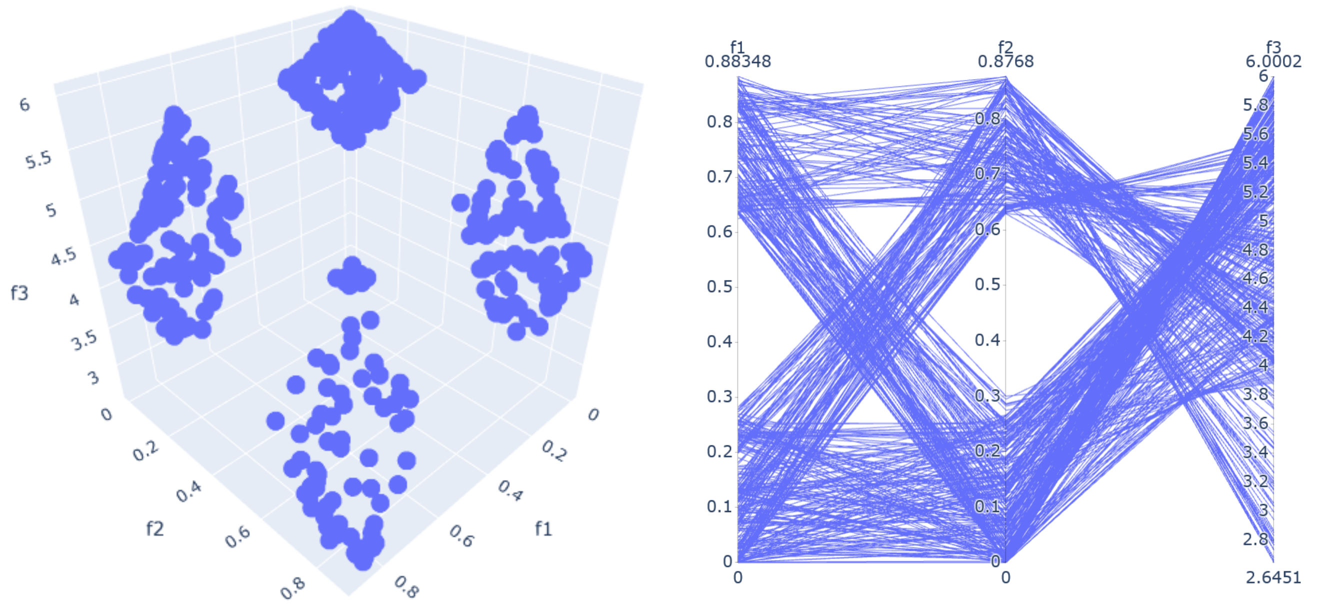
In Subsections III-A through III-D, we describe the components of the aforementioned algorithm. We also demonstrate the advantage of using the components individually with a dataset of 1036 outcome vectors generated from the DTLZ7 problem [10] with 3 objective functions. The dataset, to be referred to as (3-DTLZ7) is visualized in Figure 1a as a 3-D scatter plot and in Figure 1b as a parallel coordinate plot. Note that the SCORE band algorithm can be used to visualize outcome vectors related to problems with, at least theoretically, any number of objective functions, but we only use a three-objective problem here to describe the algorithm. We compare the results against some standard visualization techniques.

### A. OUTCOME VECTOR CLUSTERING

Clustering can be a very useful tool for a DM. Clustering can highlight patterns in the objective space by identifying groups of outcome vectors that are close to each other (i.e., in a cluster) and far from outcome vectors in other groups. In Figure 1a, the 3D scatter plot clearly shows that the outcome vectors are distributed among four clusters. However, as seen in Figure 1b, the cluster information is harder to detect in the parallel coordinate plot. While the four clusters are clearly distinguishable with regard to the first two axes, they are not nearly as distinguishable with regard to the third. However, this problem is alleviated by giving colors to the individual outcome vectors according to their cluster. Figure 2 demonstrates the effectiveness of this in the parallel coordinate plot shown. It now becomes trivial to distinguish the four clusters, even in the case of the third objective function. The spread of each cluster is visible along each axis. We can cluster outcome vectors based upon their components using a wide variety of clustering algorithms. We mention some of them in Appendix B.

### B. AXIS ORDERING

The order of the objective functions in a parallel coordinate plot can make a significant difference in interpretation. An axis in a parallel coordinate plot can have at most two neighboring axes. Hence, the relationship of an objective function to its neighbors is very prominent visually. The crossing over of the value paths of individual outcome vectors between neighboring axes signifies a negative correlation. If instead, the value paths are primarily parallel to each other, a DM can conclude that the two neighboring objective functions are highly positively correlated. Finally, a chaotic tangle of value paths signifies a lack of correlation between neighboring objective functions. In static visualizations, i.e., plots that a user cannot interact with or change, this information comes at the cost of a lack of information about non-neighboring objective function pairs. Dynamic visualizations can help a DM solve this problem by allowing a manual reordering of the axes in real-time, but it may be time-consuming to find the most informative order. However, static visualizations are still necessary for certain media where dynamic visualizations are impossible, such as in print. Hence, methods to most informatively order the axes in a parallel coordinate plot to highlight information relevant to a DM are still required.




(a) 3D scatter plot (b) Parallel coordinate plot

FIGURE 1: Dataset (3-DTLZ7) visualized using two techniques.

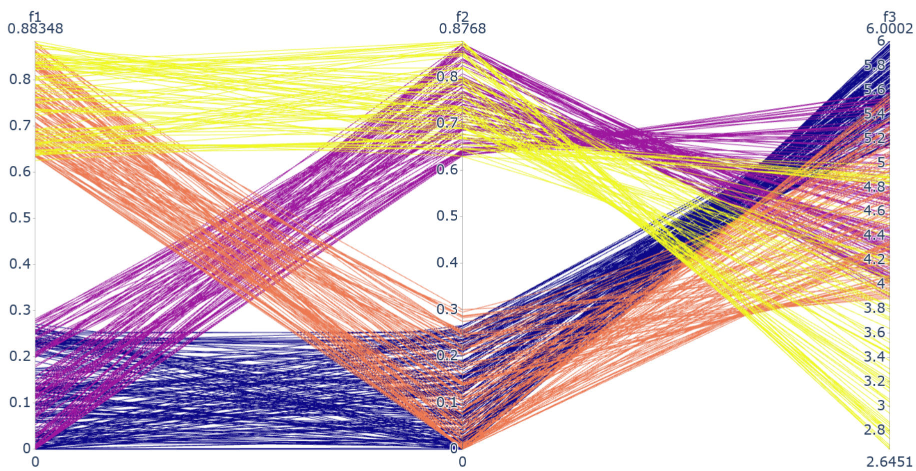



FIGURE 2: Dataset (3-DTLZ7) plotted in a parallel coordinate plot and colored according to clustering information.

As mentioned earlier, many techniques to find an order of objective functions have been proposed in the literature. However, we found the order proposed by such methods suboptimal for usage in SCORE band visualizations. We therefore propose a new way to find a good order for the objective functions.

We derive the order of the objective functions (represented by a permutation $\pi$) by solving a travelling salesperson problem $TSP(c_{i,j})$, where $c_{i,j} \in \mathbb{R}$ is a measure of the distance between two objective functions. The ordering is given by a permutation $\pi$ of the set $\{1, 2, ..., k\}$, where $k$ is the number of objective functions, such that $f_{\pi_i}$ is the $i^{th}$ objective function to be placed as an axis in a parallel coordinate plot [3]. We provide two options for the distance metric. In the first option, referred to as *Metric 1* in the following, $c_{i,j} = -\rho(f_i, f_j)$, where $\rho(f_i, f_j)$ is the Pearson correlation coefficient between $f_i$ and $f_j$. By using this metric, objective functions that are positively correlated are placed closer to each other. Alternatively, we can use the absolute value of the Pearson correlation coefficients instead: $c_{i,j} = -|\rho(f_i, f_j)|$. This second option is referred to as *Metric 2*. Metric 2 highlights both positive and negative correlations. In either case, the Pearson correlation

[3] Note that $TSP(c_{i,j})$ is not to minimize the total distance of a round-trip of the objective functions. Instead, it is only to find the shortest path that visits each objective function once.

coefficients are calculated using of all the outcome vectors that are to be visualized.

Figure 3 shows the effectiveness of using the axis-ordering method described above on the (3-DTLZ7) dataset. The second metric, which highlights both positive and negative correlations, was used to generate the figure. One can immediately notice the visual symmetry in the figure across the $f_3$ axis. The symmetry corresponds to a similar symmetry seen in the 3D scatter plot visualization (Figure 1a) around the $f_3$ axis. On the other hand, the standard parallel coordinate plot (Figure 1b) obscures this symmetry to some extent.

In static visualizations, the axis-ordering method can help the DM gain insight into the problem swiftly. In dynamic visualizations, the method can provide a default first view, which can then be altered by a DM interactively as desired.

### C. INTER-AXIS SPACING

A standard parallel coordinate plot dedicates an equal amount of space to each neighboring objective function axis pair. However, in reality, the information that a DM can gather from inter-axis spacing can vary significantly between different objective function pairs. For example, some objective function pairs may be highly correlated, whereas others may be hardly correlated at all. Thus, we can encode relevant information by altering the spacing between axes. For example, by varying the relative distances between the objective function axes, we can visually show objective function clusters: objective functions that behave similarly are placed closer to each other, whereas objective functions that behave dissimilarly are placed farther apart.

We provide two different methods for calculating relative distances between neighboring axes ($\text{dist}_i = \text{dist}(f_{\pi_i}, f_{\pi_{i+1}})$):

$$\begin{aligned} \text{Method 1:} \quad & \text{dist}_i = 1 - \rho(f_{\pi_i}, f_{\pi_{i+1}}) + \delta \\ & \text{for all } i = 1, \ldots, k-1 \\ \text{Method 2:} \quad & \text{dist}_i = \frac{1}{|\rho(f_{\pi_i}, f_{\pi_{i+1}})| + 1} + \delta \\ & \text{for all } i = 1, \ldots, k-1, \end{aligned}$$

where $\delta$ is an analyst-provided distance parameter[4] which increases the minimum distance between the axes. Based on our experiments, we recommend Method 1 be used with Metric 1 (to calculate the axes order), and Method 2 be used with Metric 2. These relative distances can be multiplied by a scaling factor to fit a desired width, for example, the width of a monitor or a page.

Axis ordering and inter-axis spacing work in tandem to help the DM study the correlations among the objective functions. The first makes it possible to depict a high/low correlation (neighboring objective functions have a high correlation, whereas non-neighboring objective functions have a lower correlation). The second adds a measure of the degree of correlation (highly correlated neighboring objective functions are placed closer to each other compared to neighboring objective functions with a lower correlation). Note that due to the symmetrical nature of the DTLZ7 problem, this aspect of SCORE band visualizations cannot be recognized in Figure 3. However, we present more problems in the later sections of the paper, which demonstrate the utility of varying inter-axis spacing.

### D. VISUALIZATION VIA BANDS

As mentioned, parallel coordinate plot visualizations tend to grow complicated and cluttered with an increasing number of objective functions and outcome vectors. As the number of objective functions increases, the outcome vector value paths cross-over more often due to the trade-offs among the different objective functions. On the other hand, adding more outcome vectors to the plot increases the complexity by simply increasing the density of information in the visualization. Together, this can result in visualizations that are difficult to understand even with the helpful features described in the previous subsections.

One way to resolve this issue is to plot simplified abstractions rather than all individual outcome vectors. For example, instead of plotting all outcome vectors as individual value paths, the clusters (as identified in Subsection III-A) can be plotted as bands as the basic unit of visualization. In Figure 4, we showcase this idea by plotting the (3-DTLZ7) dataset. We call the result a SCORE band visualization. Each band exemplifies the pattern of the trade-offs followed by the outcome vectors of the corresponding cluster while keeping the visualization simple. The width of the band at any axis represents the spread of corresponding objective function values achieved by the outcome vectors belonging to the cluster. The height, width, and shape of the bands can be calculated in various ways.

We propose that the center of a band (on each axis) be placed on the median value achieved by the outcome vectors belonging to that cluster. To determine the width of each of the bands along each axis, we can use a statistical measure of spread, such as standard deviation, interquartile range, or confidence interval. We use an interquartile range achieved by the outcome vectors in each cluster for each objective function to generate the bands in Figure 4, referred to as "50% bands"[5] in the legend. Once we have the height (cluster median) and width (cluster interquartile range) at each axis, we draw the band by interpolating between the axes. A traditional parallel coordinate plot does this by linear interpolation, leading to polyline value paths for each outcome vector. However, we have found that using spline interpolation leads to a more aesthetically pleasing visualization without extraneous informa-

[4]Depending upon the correlation values, a small value of $\delta$ may lead some objective functions to be placed too closely together, making the visualization difficult to interpret. In such cases, the analyst can increase $\delta$ to desirable levels. Setting a very large value ($> 20$) for $\delta$ will lead to equidistant objective function axes.

[5]The term "50% bands" refers to the fact that the interquartile range contains 50% of the objective vectors. Note that we use the term "solutions" in the legend of the visualization instead of "outcome vectors" for compactness, and use the available space to show the number of outcome vectors in each cluster.

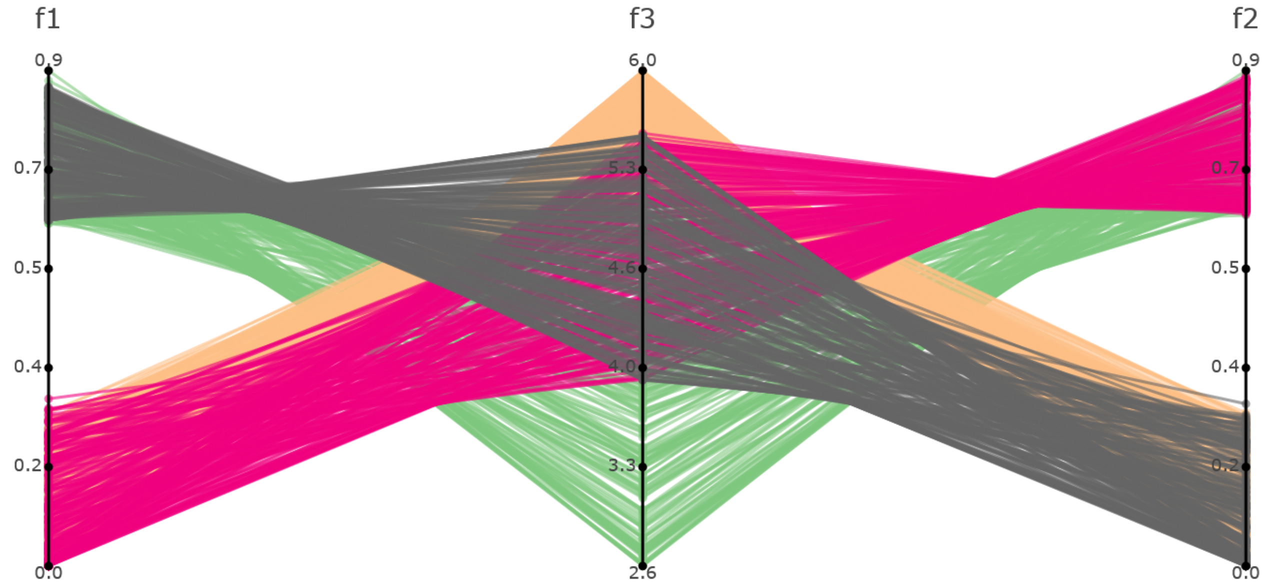



FIGURE 3: Dataset (3-DTLZ7)plotted in a parallel coordinate plot, colored according to the clustering information, and with axes ordered according to Metric 2, i.e., the absolute values of the Pearson correlation coefficients.

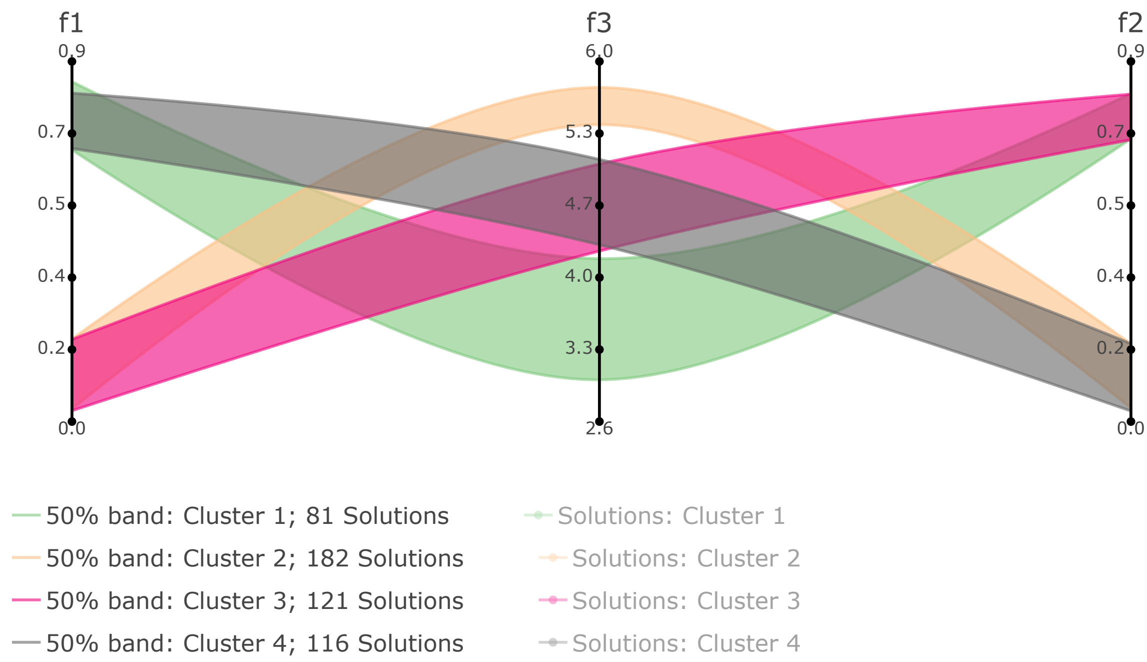



FIGURE 4: Dataset (3-DTLZ7) plotted as a SCORE band visualization.

tion[6]. We color the bands according to clustering information and make them translucent to make it easier to distinguish between the different clusters and follow their patterns.

Note that a SCORE band visualization is meant to be the first or default visualization to be used in a decision making process. No single view is able to tell all relevant information, hence, a DM must be able to affect the appearance in an interactive manner to decide whether they want to see more or less details. The bands make it easy for a DM to

[6]Our implementation uses a Catmull-Rom spline interpolation.

identify patterns in the outcome vectors and the distances between axes communicate correlation information. As we have shown, it may otherwise be challenging to gain insight and find essential information from a large amount of data. SCORE bands provide support in analyzing outcome vectors gradually: observing main trends first and then deciding how to proceed and concentrating on outcome vectors that are of interest.

For example, the DM can study SCORE bands one by one, eliminate bands that are not acceptable or identify a band or some bands of interest and concentrate on them to see individual outcome vectors corresponding to selected SCORE bands. As an example, the DM can select one SCORE band as a region of interest and selectively visualize and filter out outcome vectors in it until a most preferred solution is found.

Alternatively, based upon the insight gained, a DM can, for example, give their preferences for the next step of an interactive decision making process. Depending on the interactive method employed, the preference information can be as little as one SCORE band. Or the DM can provide preferences for clusters of objective functions as opposed to all objective functions. SCORE bands can also be used before an interactive decision making process to give an overview of a pre-generated rough representation of Pareto optimal outcome vectors. It is then easier for the DM to provide preferences when an overview of achievable outcome vectors is known.

We have implemented SCORE band visualization as an open-source Python package and will make the code available. The package creates interactive visualizations which support both SCORE bands and traditional parallel coordinate plots for studying individual outcome vectors. In the interactive visualization, a DM can show or hide various bands or clusters of outcome vectors corresponding to the bands by interacting with the legend of the plot. For example, in Figure 4, the legend entries for outcome vectors (“Solutions: Cluster 1” – “Solutions: Cluster 4”) appear translucent compared to the legend entries for the four bands. This signifies that individual outcome vectors are hidden. A DM can make those outcome vectors visible by clicking on the corresponding legend entries. Additionally, the DM can also hide one or more of the bands by clicking on their legend entries. We denote all parameter settings that control the appearance and behavior of SCORE band visualization and their default values in Appendix B. Note that a DM is not expected to change these settings; an analyst assisting the DM can do so, if required.

## IV. CASE STUDIES

In this section, we demonstrate the usage of SCORE band visualizations with a variety of datasets (i.e., sets of outcome vectors). These include generated datasets, datasets obtained from multiobjective optimization benchmark problems, and real-life multiobjective optimization problems. We share all the datasets at https://doi.org/10.5281/zenodo.14025276. We showcase how the various parameters of the SCORE band visualizations can be changed to highlight different aspects of the explored datasets. We also show how supporting visualizations can help a DM understand the SCORE band visualizations and the data.

### A. ARTIFICIAL DATASETS

We begin by visualizing a small artificial dataset (AD1) consisting of 11 outcome vectors with six components with a known correlation structure described in Appendix A. We visualize (AD1) using a standard parallel coordinate plot in Figure 5. The figure shows that there are vectors in clusters, but the exact number of clusters is not immediately clear. Additionally, as parallel coordinate plots are not designed to display information related to the correlation of the objective functions, that information is lost in this visualization.

Figure 6 visualizes Dataset (AD1) using SCORE bands. We used Variational Bayesian Gaussian mixture to calculate the outcome vector clusters and Method 1 to determine inter-axis spacing. The three clusters of outcome vectors can be seen as three bands of different colors. The clusters of objective functions are also clearly visible in the pairs: (f2, f1), (f3, f4), (f6, f5). The objective functions that are closer to each other (f1 and f2, for example) have a very low degree of “crossing over” of bands, signifying a high correlation. On the other hand, the bands “cross over” much larger (vertical) distances when the neighboring objective functions are farther apart (such as f1 and f3), signifying a negative correlation.

The second dataset (AD2) mentioned in Appendix A has three clusters of objective functions with high in-group correlations (consisting of 2, 3 and 4 objective functions, respectively). While creating the dataset, we ordered the objective functions such that consecutive objective functions (for example, f1 and f2, or f5 and f6) are negatively correlated. As seen in Figure 7, such datasets can be particularly challenging to interpret using a standard parallel coordinate plot.

We visualize the dataset (AD2) using SCORE bands in Figures 8a and 8b. We used the same parameter values for the visualization as for (AD1). Figure 8a shows the dataset in the form of bands of clustered outcome vectors, whereas Figure 8b hides the bands and shows the outcome vectors directly. The three clusters of objective functions are clearly visible (with 4, 2 and 3 objective functions forming the three clusters). As mentioned earlier, neighboring objective functions that are placed closer to each other have high correlations. As a consequence of this, a DM can simultaneously improve objective functions belonging to such clusters without much compromise. We see this in Figure 8b with objective functions f5, f3, and f7. Most of the outcome vectors (especially those belonging to cluster 2 (black)) have value paths that are nearly parallel to each other, with minimal crossing over. We can improve these three objective functions simultaneously (at the cost of other objective functions). Hence, by ordering and spacing axes according to Pearson correlation coefficients, we simplify the decision making process.

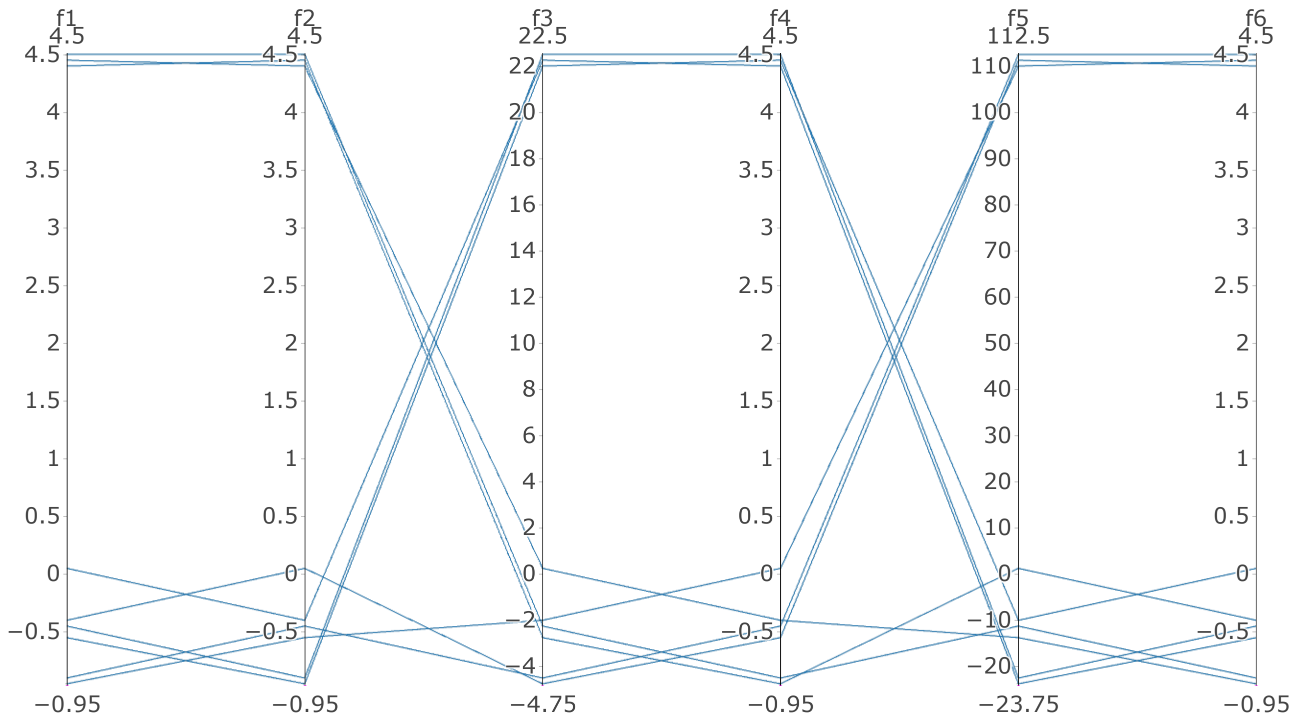



FIGURE 5: Visualizing the (AD1) dataset using a parallel coordinate plot

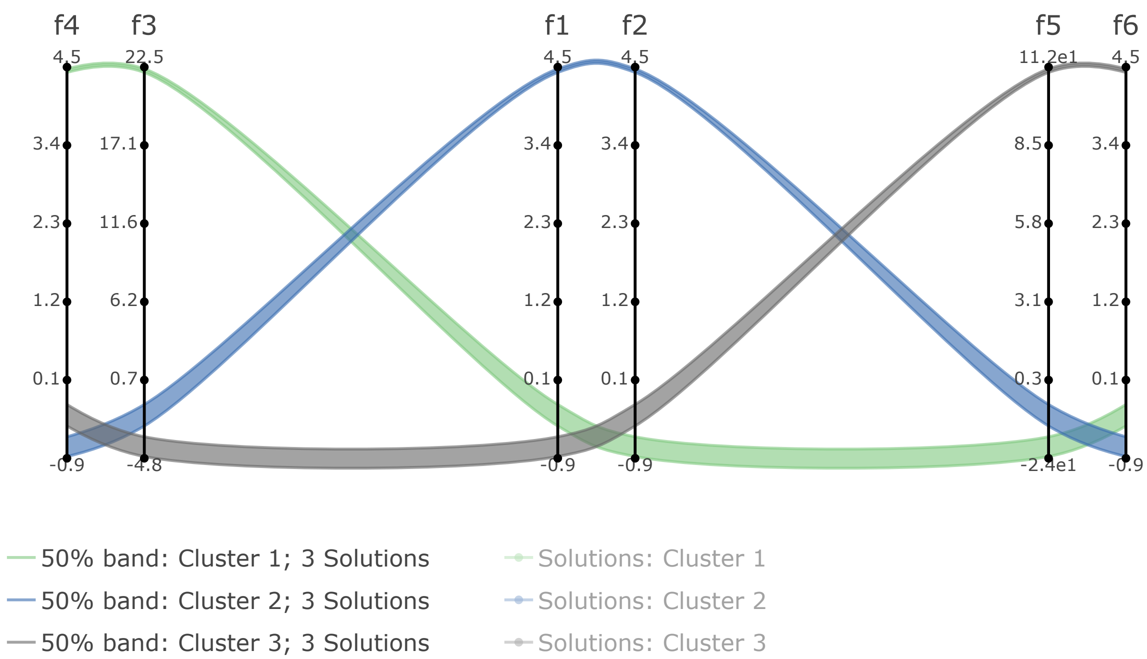



FIGURE 6: Visualizing the (AD1) dataset using SCORE band visualization

Figure 8b is further simplified by considering the bands in Figure 8a. As (AD2) did not have clustered outcome vectors,

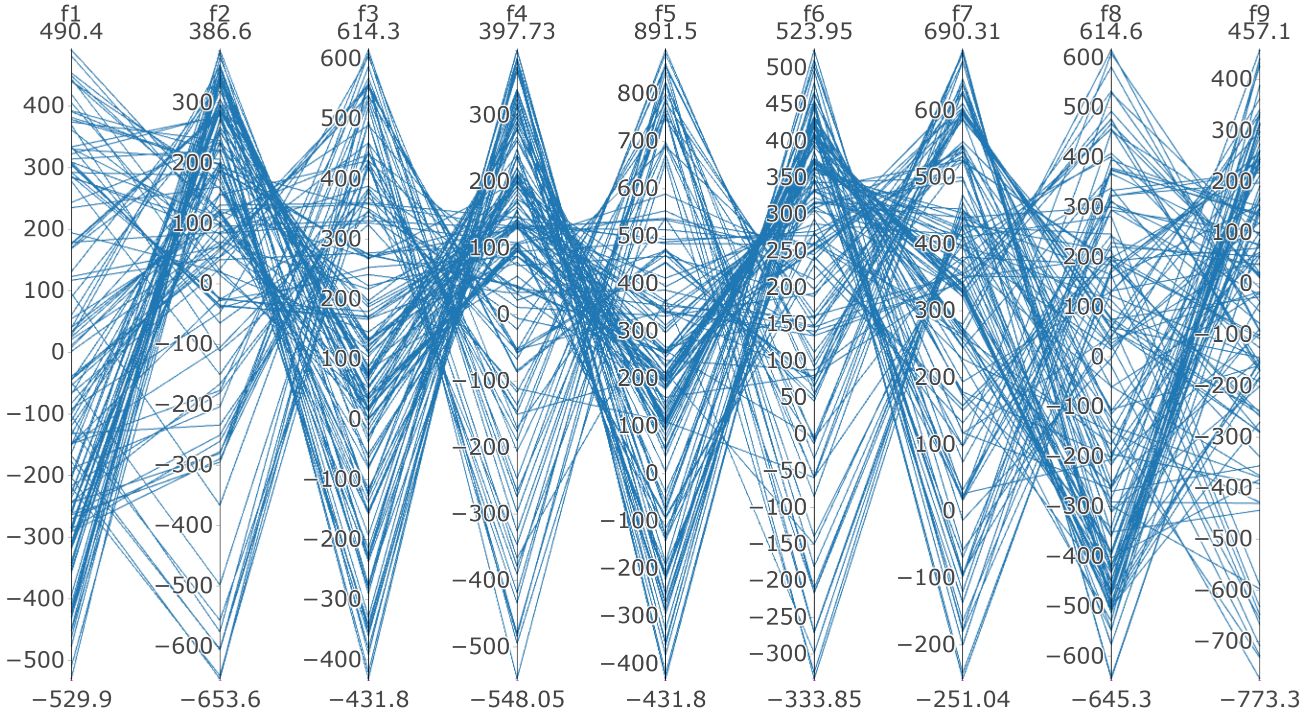



FIGURE 7: Visualizing the (AD2) dataset using a parallel coordinate plot

the clusters and bands created for Figure 8a may be misleading. Some DMs may still prefer Figure 8a as a starting point for decision making. The bands provide simple abstractions for the patterns followed by groups of outcome vectors (regardless of whether the clusters genuinely exist or not). Once a DM identifies a region of interest using the bands, we can hide the bands and show individual outcome vectors. We discuss this aspect further in the following subsection.

The (AD2) dataset also showcases how SCORE bands tackle global and local correlations (or trade-offs). The axes are ordered and spaced using global Pearson correlation coefficients, and therefore only reflect the global trade-offs. This leads to the behavior in Figure 8b where outcome vectors that, e.g., attain a high value for one of the objective functions (f2) also attain a high value for the other objective functions in the same cluster (f9, f6, and f4). As mentioned earlier, this leads to value paths being predominantly parallel to each other.

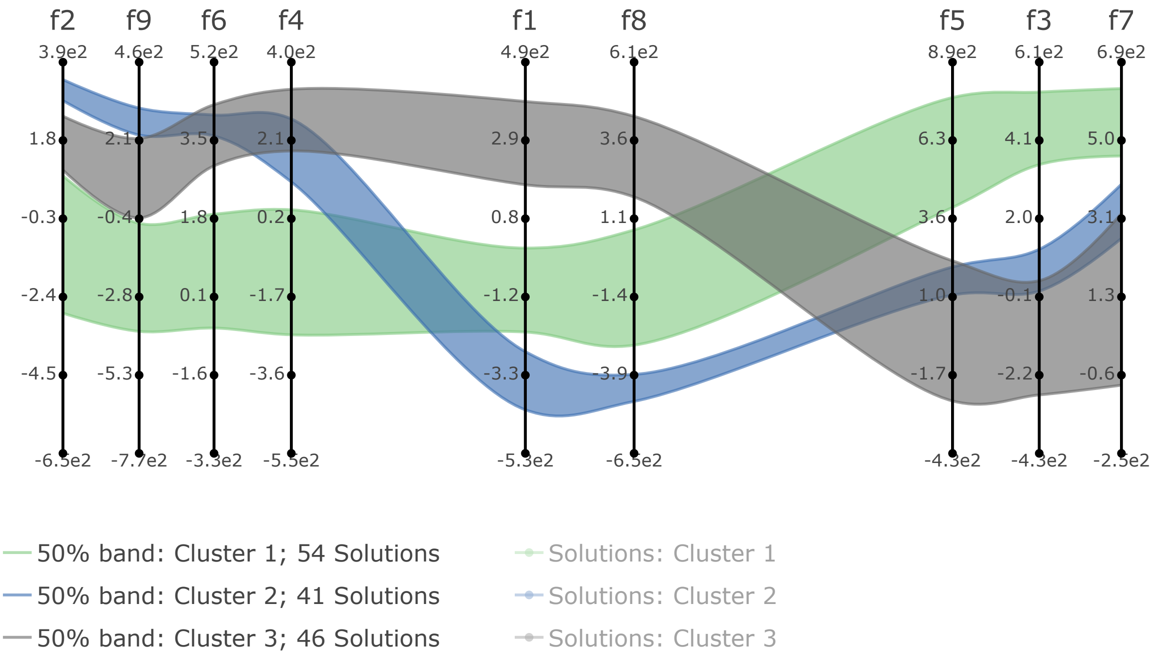
However, if local correlations do not follow the global patterns, the value paths may still intersect each other between closely placed (hence highly correlated) axes. This can be seen in Figure 8b in the second cluster of axes (f1 and f8). The corresponding objective functions are highly correlated on a global scale: the black cluster attains high values for both objective functions, the green cluster attains average values, whereas the blue cluster attains a low value for both objective functions. However, within each cluster, there is a large amount of crossing over. For example, the outcome vectors in the green cluster with low values for the objective function f1 achieve a high value for the objective function f8.

SCORE bands get around this issue by allowing the DM to conduct decision making in steps. First, they can look at the visualization with bands, Figure 8a, and identify their region of interest at a global scale. Instead of focusing on nine objective functions and hundreds of outcome vectors, they can focus on three clusters of objective functions (shown by spacing between axes), and three clusters of outcome vectors (shown via coloured bands). Once they have identified a band of interest, they can visualize outcome vectors only within that band, hiding everything else. Then, they can focus on the local trade-offs and choose the most preferred outcome vector.

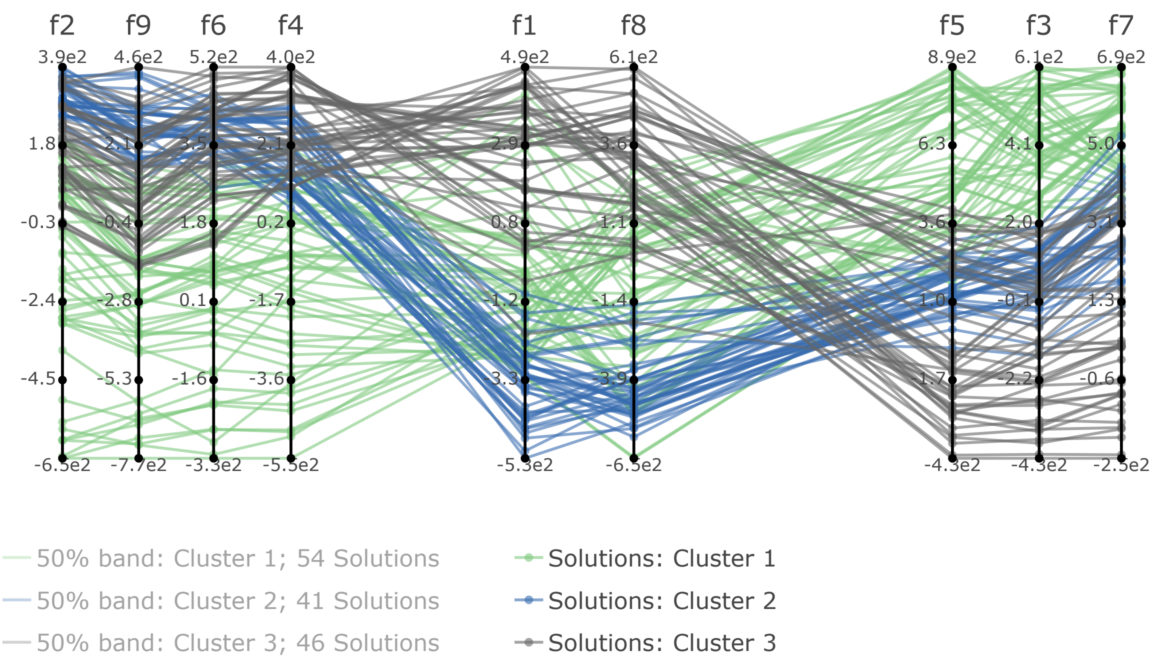
### B. BENCHMARK DATASET

We show the effectiveness of using clustering algorithms even with datasets in which no real clusters exist using the DTLZ5 benchmark problem with a degenerate Pareto front. We consider 3 objective functions and denote the set of 1000 Pareto optimal outcome vectors by (3-DTLZ5). Figures 9 and 10 show (3-DTLZ5) as SCORE band visualizations. For generating Figure 9, we used the DBSCAN algorithm to generate the outcome vector clusters. As there are no clusters in the dataset, the visualization puts all outcome vectors in a single cluster. Hence, the resulting singular band does not provide much information to the DM, and it is necessary to display individual outcome vectors, instead.

Figure 10, on the other hand, uses Gaussian mixture models




(a) SCORE band visualization




(b) Visualization of individual outcome vectors

FIGURE 8: Visualizations of the (AD2) dataset

for clustering. The algorithm forcibly breaks the dataset down into many clusters of similar sizes. These clusters group

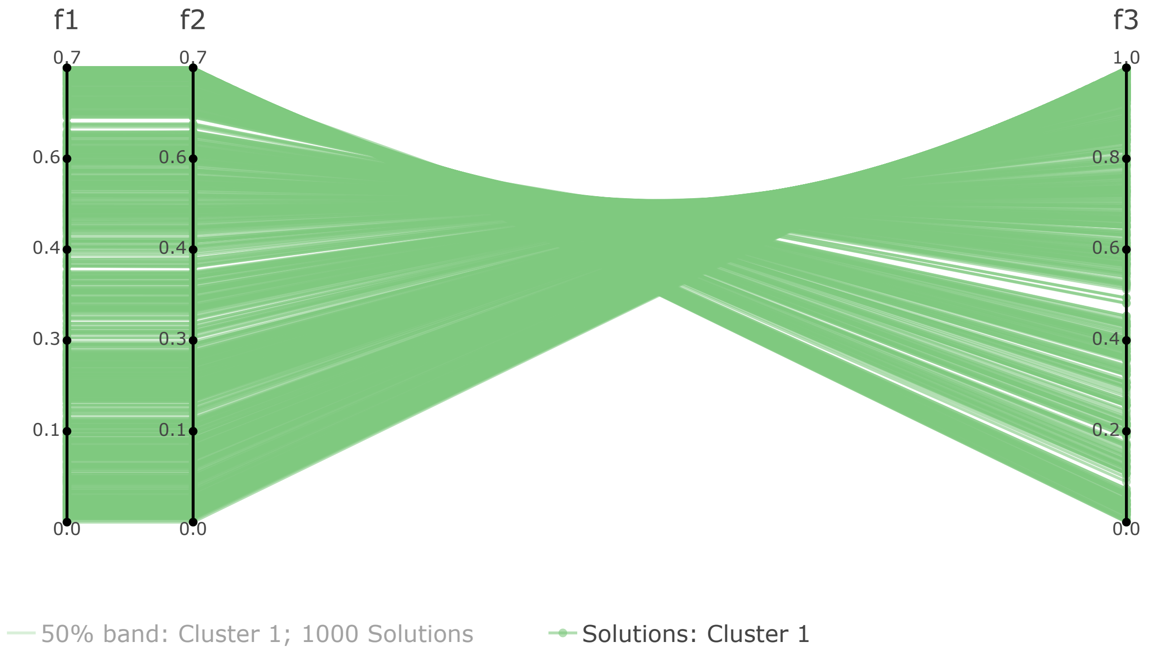



FIGURE 9: SCORE band visualization of the (3-DTLZ5) dataset using DBSCAN as the clustering algorithm (the band is hidden and just individual outcome vectors are shown).

outcome vectors that are close to each other in objective space, and individual clusters lie adjacent to other clusters along the degenerate curve of the Pareto front. These clusters are created randomly, and running the clustering algorithm multiple times returns slightly different groupings (both in the number and members of clusters). While the individual clusters have no real significance, they provide a simpler way of understanding patterns in the dataset. Instead of focusing on hundreds or thousands of outcome vectors as in Figure 9, a DM can focus on a much smaller number of bands. Information about the trade-offs and correlations can easily be understood by following the paths of the bands and comparing the relative distances between the axes, respectively. It is also easier to compare a small number of bands to discover a region of interest than doing the same in a plot with thousands of outcome vectors. When a DM finds such a region, they can focus on the outcome vectors belonging to the clusters in the region and hide all other outcome vectors/bands, effectively ‘‘zooming’’ into the region of interest.

### C. REAL-LIFE DATA-BASED PROBLEMS

To demonstrate the usage of SCORE band visualizations in real-life problems, we use the general aviation aircraft design (GAA) problem [40]. We use 709 Pareto optimal outcome vectors for the problem obtained in Mazumdar, Chugh, Hakanen, *et al.* [30] with eleven objective functions. We visualize this (GAA) dataset using SCORE bands in Figures 11 and 12 and present a supporting visualization in Figure 13 in the form of a heatmap of pairwise Pearson correlation coefficients of the problem's eleven objective functions.

Figure 11 uses DBSCAN as the clustering algorithm for (GAA), whereas Figure 12 uses Gaussian mixture models. In both figures, it is immediately apparent that there are three groups of objective functions: two groups with high in-group correlations ({f1, f2, f3, f5, f6} and {f7, f8, f9}) and a third group with lower in-group correlations ({f4, f10, f11}). We can verify this property by looking at the pairwise correlations in Figure 13 that shows a heatmap of the Pearson correlation coefficient values among the different objective functions. Note that the Pearson correlation coefficients can also help explain the order of objective functions in a SCORE band visualization and, in turn, may enable a DM to manually decide the order of objective functions in parallel coordinate plots.

Furthermore, by following the value paths of one of the bands (for example, the gray band for ‘‘Cluster 4" in Figure 11, which has the lowest f2 objective function value), it is clear that there is no or minimal in-group trade-off between the objective functions of the first two groups. On the other hand, there are significant out-group trade-offs between the objective functions of the same two groups. Such a behavior in the data makes the decision making process significantly simpler as instead of focusing on the trade-offs among **eleven** objective functions, a DM can focus on the first **two** groups

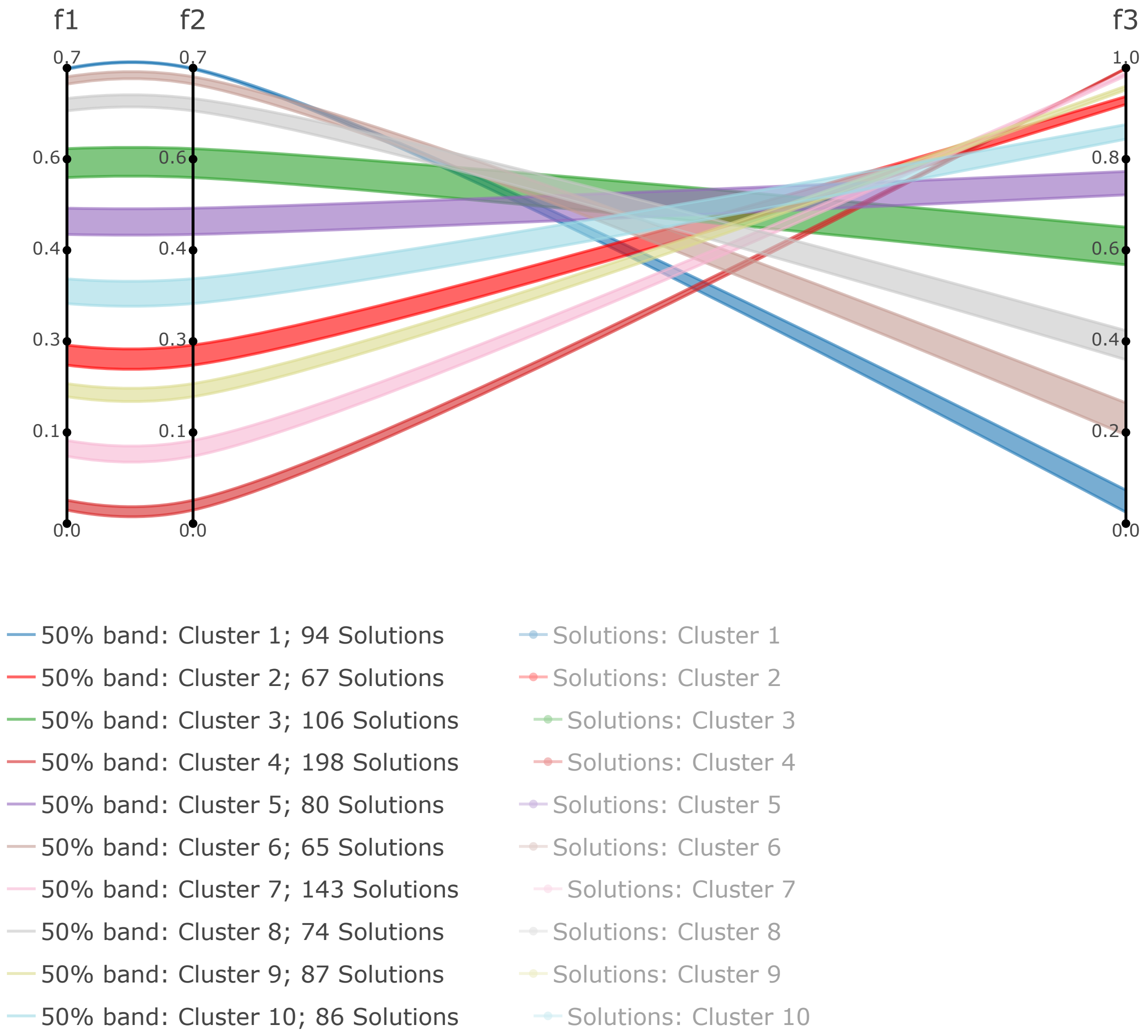



FIGURE 10: SCORE band visualization of the (3-DTLZ5) dataset using Gaussian mixture models as the clustering algorithm.

and the remaining **three** objective functions.

Even though Figures 11 and 12 visualize the same dataset, they look significantly different because of the behavior of the corresponding clustering algorithms. Using the DBSCAN algorithm results in a very simplified plot, which reduces the time taken by a DM to visually gather the information needed to understand simple patterns in the data. However, this simplicity hides the outcome vectors that can clearly be seen in Figure 12. While this is a major downside in static visualizations, we solve this problem by making the plots interactive and enabling the DM to visualize all outcome vectors from one or more clusters at any time. In this way, the DM can iteratively narrow down the search for a most preferred SCORE band – and finally to a most preferred outcome vector – using the available information on correlated and independent objective functions, respectively. If, for example, a small value in objective function $f_5$ is important to the DM, one can observe that this comes with small objective function values in the correlated objective functions $f_2, f_6, f_3, f_1$ and also $f_4$, however, at the cost of large objective function values in the objective functions $f_9, f_7, f_8$ that are in conflict with this preference. Clusters 7 or 9 may be the most preferred clusters in this case. Then, the DM can zoom-in on the desired SCORE band and focus on its corresponding outcome vectors (instead of comparing all outcome vectors at the same time).

We cannot predict the behavior of the clustering algorithms, as the results are dependent on both the nature of the objective functions and the distribution of outcome vectors to be visualized (discovered by the optimization algorithm). Hence, we recommend that a DM should be presented with at least two different SCORE band visualizations using different clustering algorithms.

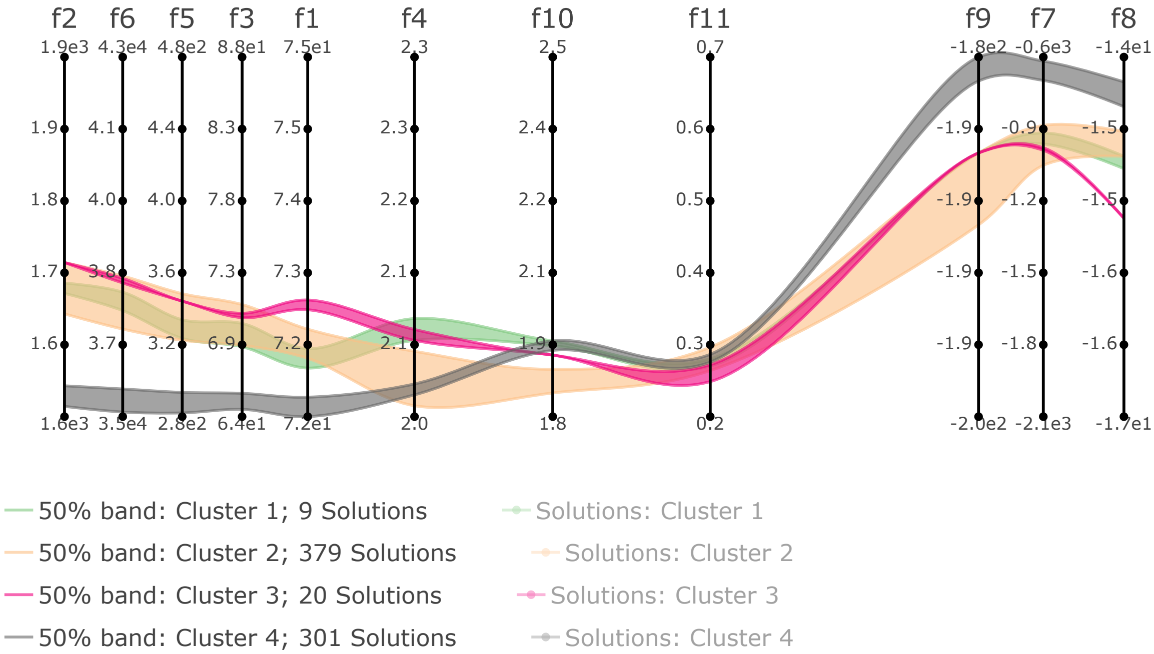



FIGURE 11: SCORE band visualization of the (GAA) dataset using DBSCAN as the clustering algorithm

## V. CONCLUSIONS

In this paper, we have proposed SCORE band visualizations as a novel way of presenting the outcome vectors of a multiobjective optimization problem with many objective functions to a decision maker. The proposed technique explicitly addresses the case of many objective functions in which it becomes difficult for the decision maker to process the outcome vectors generated. While different methods generate varying amounts of outcome vectors, the task of evaluating this information, all other things equal, only gets more demanding as the number of objective functions increases.

Already in the three-objective case, a direct visualization of outcome vectors in the objective space is difficult, and it is generally not useful at all in cases of more than three objective functions. Hence, there is a growing need for economical and meaningful presentations of Pareto optimal or nondominated outcome vectors.

SCORE bands support decision making by identifying patterns in the information and breaking it down into digestible components to make it easier to gain major insights. The decision maker can easily grasp correlated objective functions since they are arranged next to each other. Moreover, in contrast to common techniques, we vary the distance among objective functions to transmit the degrees of correlation. By visually grouping together highly correlated objective functions, we make it easy to identify objective functions that behave similar to each other. By representing clusters as simple-to-distinguish bands and hiding individual outcome vectors by default, we reduce the cognitive burden on a DM. Instead of requiring a DM to choose an outcome vector from many in a cluttered plot, we present a visually simple plot, and ask them to choose a band from a small number of bands. Once they select a band, they can make the outcome vectors belonging to the same cluster visible by interacting with the plot, and finally choose their most preferred outcome vector.

Our aim is to provide assistance by means of an intelligent way of visualization which is applicable to problems possessing multiple conflicting objective functions. Our future research directions include integrating SCORE bands in an interactive decision making process. This will allow us to utilize these visualizations to provide preference information to interactive algorithms such as those implemented in the DESDEO software framework. This will enable us to conduct case studies (with, for example, students and real-world decision makers) and analyze how decision makers use this tool to solve real-world problems.

## ACKNOWLEDGEMENTS

The idea of this research was developed at the Dagstuhl Seminar 20031 (Scalability in Multiobjective Optimization). The research is related to the thematic research area DEMO (Decision Analytics utilizing Causal Models and Multiobjective Optimization, jyu.fi/demo) of the University of Jyvaskyla. We would like to thank Dr Johanna Silvennoinen for helping with research related to the field of cognitive science as well as Dr Giovanni Misitano for providing feedback during the writing

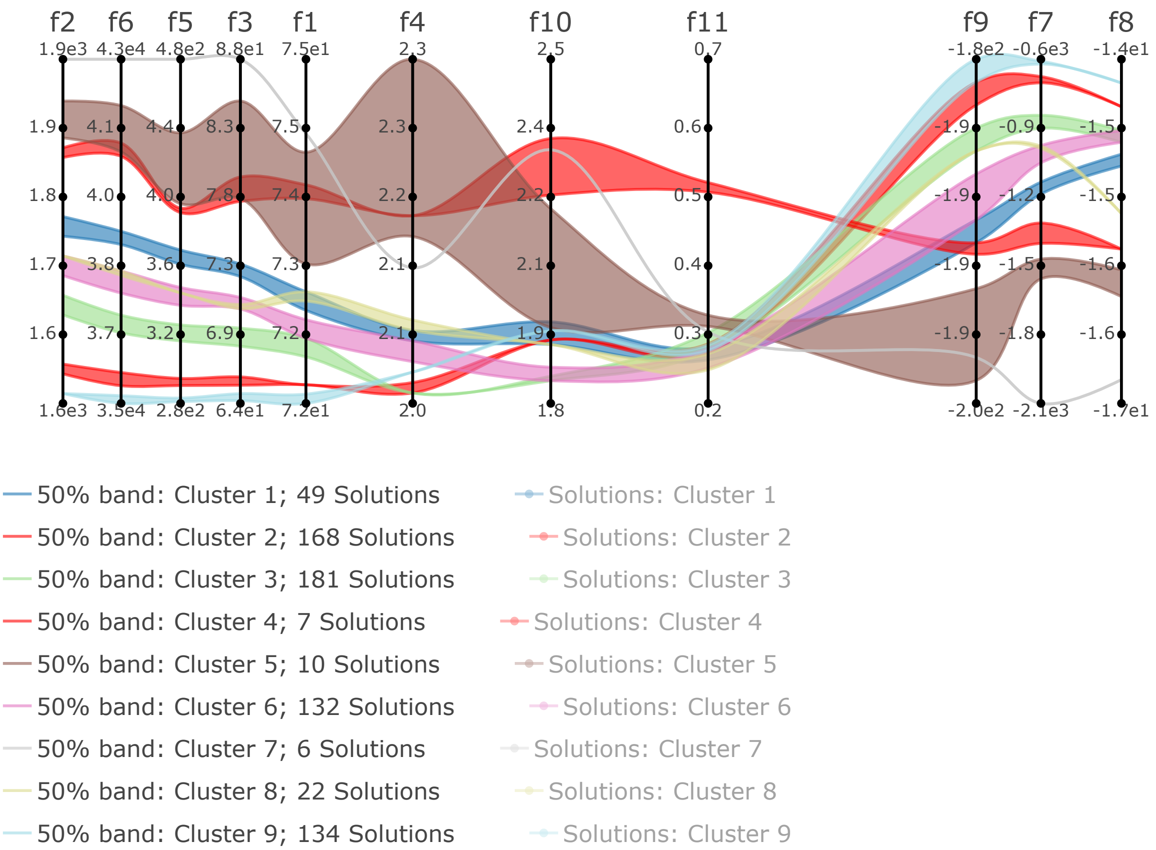



FIGURE 12: SCORE bands of the (GAA) dataset using Gaussian mixture models as the clustering algorithm

| | f2 | f6 | f5 | f3 | f1 | f4 | f10 | f11 | f9 | f7 | f8 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| f2 | 1.0 | 1.0 | 0.99 | 0.98 | 0.95 | 0.67 | 0.2 | 0.16 | -0.96 | -0.98 | -0.89 |
| f6 | 1.0 | 1.0 | 0.99 | 0.99 | 0.96 | 0.7 | 0.25 | 0.18 | -0.96 | -0.98 | -0.89 |
| f5 | 0.99 | 0.99 | 1.0 | 0.99 | 0.94 | 0.66 | 0.2 | 0.14 | -0.97 | -0.99 | -0.87 |
| f3 | 0.98 | 0.99 | 0.99 | 1.0 | 0.96 | 0.71 | 0.27 | 0.23 | -0.95 | -0.98 | -0.89 |
| f1 | 0.95 | 0.96 | 0.94 | 0.96 | 1.0 | 0.74 | 0.32 | 0.28 | -0.88 | -0.94 | -0.91 |
| f4 | 0.67 | 0.7 | 0.66 | 0.71 | 0.74 | 1.0 | 0.58 | 0.33 | -0.63 | -0.65 | -0.69 |
| f10 | 0.2 | 0.25 | 0.2 | 0.27 | 0.32 | 0.58 | 1.0 | 0.48 | -0.13 | -0.26 | -0.41 |
| f11 | 0.16 | 0.18 | 0.14 | 0.23 | 0.28 | 0.33 | 0.48 | 1.0 | -0.1 | -0.15 | -0.25 |
| f9 | -0.96 | -0.96 | -0.97 | -0.95 | -0.88 | -0.63 | -0.13 | -0.1 | 1.0 | 0.96 | 0.78 |
| f7 | -0.98 | -0.98 | -0.99 | -0.98 | -0.94 | -0.65 | -0.26 | -0.15 | 0.96 | 1.0 | 0.88 |
| f8 | -0.89 | -0.89 | -0.87 | -0.89 | -0.91 | -0.69 | -0.41 | -0.25 | 0.78 | 0.88 | 1.0 |

FIGURE 13: Pearson correlation coefficient values between the objective functions in the (GAA) dataset.

process.

## APPENDIX A
## GENERATION OF TEST CASES – STRUCTURED DATASETS WITH PARTIALLY CORRELATED OBJECTIVE FUNCTIONS

We suggest two instances of multiobjective linear programming problems (MOLPs) with known partial correlations in order to validate – and illustrate – SCORE bands visualizations. Since the construction can be easily generalized, such instances can be used as test cases for other visualization methods as well. Towards this end, we consider MOLPs of the form

$$\min\{\boldsymbol{f}(\boldsymbol{x}) = C\boldsymbol{x} \;:\boldsymbol{x} \in \{A\boldsymbol{x} \leq \boldsymbol{b},\, \boldsymbol{0} \leq \boldsymbol{x} \leq \boldsymbol{u}\}\}, \quad \text{(AD)}$$

where $C \in \mathbb{R}^{k\times n}$ is the coefficient matrix for the objective functions, $A \in \mathbb{R}^{m\times n}$ the constraint matrix, $\boldsymbol{b} \in \mathbb{R}^m$ the right-hand side vector, and $\boldsymbol{u} \in \mathbb{R}^n$ the vector of upper bounds on the variable values. The problems are designed so that they naturally induce clusters of similar objective functions and of similar outcome vectors. The first instance of problem (AD) has three variables and six objective functions with the instance's coefficients specified in Table 1. A graph of the problem is given in Figure 14.

| | $x_1$ | $x_2$ | $x_3$ | | |
|---|---|---|---|---|---|
| C | -5 | 0.5 | 1 | min | |
| | -5 | 1 | 0.5 | min | |
| | 2.5 | -25 | 5 | min | |
| | 1 | -5 | 0.5 | min | |
| | 12.5 | 25 | -125 | min | |
| | 1 | 0.5 | -5 | min | |
| s.t. | 1 | 1 | 1 | $\leq$ | 1.0 |
| | 2 | 1 | 0 | $\leq$ | 1.7 |
| u | 0.8 | 0.8 | 0.8 | | |
| | | | | all vars $\geq 0$ | |

TABLE 1: Objective and constraint coefficients of the 3-variable, 6-objective instance of (AD)

This instance (AD) has 11 Pareto optimal extreme points $\{\boldsymbol{x}^1, \ldots, \boldsymbol{x}^{11}\}$. In Figure 14, they are indicated by dots. We denote the corresponding outcome vectors by (AD1). The vectors $-\boldsymbol{c}^i$, $i = 1, \ldots, 6$, in the graph illustrate the negative gradient directions of the objective functions, indicating the directions of optimization. Note that while their directions are accurate, their lengths are merely suggestive because of the differences in scale.

As seen, the six objective functions are clustered into three batches of two each, with $-\boldsymbol{c}^1$ and $-\boldsymbol{c}^2$ pointing almost down the $x_1$-axis, $-\boldsymbol{c}^3$ and $-\boldsymbol{c}^4$ pointing almost up the $x_2$-axis, and $-\boldsymbol{c}^5$ and $-\boldsymbol{c}^6$ pointing along the $x_3$-axis. This suggests that an appropriate clustering of the objective functions would be in the groups $\{f_1,f_2\}$, $\{f_3,f_4\}$, and $\{f_5,f_6\}$, while the 11 Pareto optimal extreme points should be clustered according to their geometrical closeness on the polyhedral feasible set, i.e., in the groups $\{\boldsymbol{x}^1, \ldots, \boldsymbol{x}^5\}$, $\{\boldsymbol{x}^6, \boldsymbol{x}^7, \boldsymbol{x}^8\}$ and $\{\boldsymbol{x}^9, \boldsymbol{x}^{10}, \boldsymbol{x}^{11}\}$.

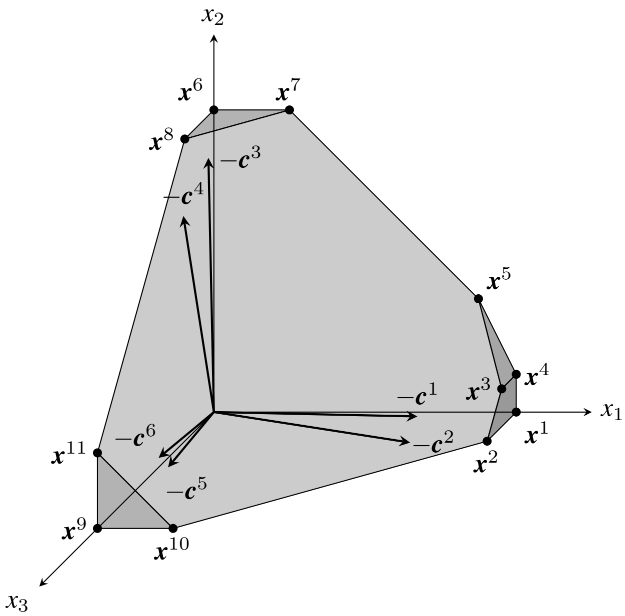



FIGURE 14: 3-variable instance (AD1) whose 6 objective functions are clustered into 3 batches of 2 each.

Our second instance of problem (AD) has nine objective functions. The corresponding dataset (AD2) has 141 Pareto optimal outcome vectors.

## APPENDIX B
## CONTROLLING THE APPEARANCE OF SCORE BANDS

We have implemented the SCORE band visualization and a graphical user interface (GUI) tool to help create the plots from data as a part of the open-source interactive multiobjective optimization framework DESDEO [34]. To enable interactive visualizations (as mentioned above), various customization options are needed (by an analyst supporting a DM) to tailor SCORE band visualizations to different needs (through the GUI):

- Outcome vector clustering: Different clustering algorithms are DBSCAN (default), Bayesian Gaussian mixture models, K-Means, spectral clustering, Ward hierarchical clustering, and agglomerative clustering. Providing the number of clusters is necessary for all algorithms except DBSCAN and Bayesian Gaussian mixture models. We have used the `scikit-learn` package to train these models [36]. We also allow the analyst to provide clustering information directly to the tool, allowing them to use clustering algorithms of their choice.
- Axis ordering: Whether to use the absolute value of the Pearson correlation coefficient as a distance metric or not. By default, Metric 1 is used.
- Inter-axis spacing: Distance function to be used (Method 1 is used by default) and the distance parameter $\delta$ (default value = 0.4).
- Visualization via bands: Display individual outcome vectors, cluster medians, and/or SCORE bands in the

plot. Each of the three options can be disabled for each cluster independently. By default, only the bands are visualized. Unlike the other options which are meant for analysts, a DM can use this option to show/hide information interactively as desired as a part of the decision making process.

We use the `Plotly` package [37] to implement the SCORE band visualizations and the related `Dash` [20] package to create a web-based GUI. Data can be imported into the GUI. The visualization can then be created without further input. Alternatively, the GUI provides a form to change the various parameters of a SCORE band visualization. We show the GUI implemented in Figure 15. Additionally, we use the `NumPy` [15] and `Pandas` [39] Python packages for data handling. We use the `SciPy` Python package [46] to extract statistical information (such as Pearson correlation coefficients) from the data. Finally, we use the `tsp_solver2` package to solve the TSP problem using a greedy algorithm [41].

While we provide default hyperparameter settings, it should be recognized that no such default can be optimal for all datasets to be visualized. For example, the figures generated to discuss the case studies use slightly different hyperparameters (as reported). The hyperparameters were chosen based on visual inspection of the results. Consequently, an analyst assisting the DM plays an important role. To help analysts with this task (apart from providing good defaults for the hyperparameters), we have devoted special focus on optimizing the provided implementation. Each time an analyst changes the parameters, they can see the results within a few seconds, even for datasets with a large number of objective functions and outcome vectors.

## References


[1] D. F. Andrews, "Plots of high dimensional data," *Biometrics*, vol. 28, no. 1, pp. 125–136, 1972.

[2] M. Ankerst, S. Berchtold, and D. Keim, "Similarity clustering of dimensions for an enhanced visualization of multidimensional data," in *Proceedings of the IEEE Symposium on Information Visualization*, IEEE, 1998, pp. 52–60.

[3] S. Bandaru, T. Aslam, A. Ng, and K. Deb, "Generalized higher-level automated innovization with application to inventory management," *European Journal of Operational Research*, vol. 243, no. 2, pp. 480–496, 2015.

[4] S. Cajot, N. Schüler, M. Peter, A. Koch, and F. Maréchal, "Interactive optimization with parallel coordinates: Exploring multidimensional spaces for decision support," *Frontiers in ICT*, vol. 5, article 32, 2019.

[5] H. Chernoff, "The use of faces to represent points in K-dimensional space graphically," *Journal of the American Statistical Association*, vol. 68, no. 342, pp. 361–368, 1973.

[6] L. Cibulski, T. May, J. Schmidt, and J. Kohlhammer, "Compo*sed: Composite parallel coordinates for co-dependent multi-attribute choices," *IEEE Transactions on Visualization and Computer Graphics*, vol. 29, no. 10, pp. 4047–4061, 2023.

[7] J. L. Cohon, *Multiobjective Programming and Planning*. New York: Academic Press, 1978.

[8] O. Da Silva, N. Crilly, and P. Hekkert, "Maximum effect for minimum means: The aesthetics of efficiency," *Design Issues*, vol. 32, no. 1, pp. 41–51, 2016.

[9] K. Dächert, K. Klamroth, K. Miettinen, and R. Steuer, "KaKaRaKe – user-friendly visualization for multiobjective optimization with high-dimensional objective vectors," in *Scalability in Multiobjective Optimization, Report from Dagstuhl Seminar 20031*, C. Fonseca, K. Klamroth, G. Rudolpth, and M. Wiecek, Eds., Dagstuhl, 2020, pp. 97–103.

[10] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scalable test problems for evolutionary multiobjective optimization," in *Evolutionary Multiobjective Optimization. Advanced Information and Knowledge Processing*, A. Abraham, L. Jain, and R. Goldberg, Eds., Springer, 2005, pp. 105–145.

[11] Y. Fang, Q. Liu, M. Li, Y. Laili, and D. T. Pham, "Evolutionary many-objective optimization for mixed-model disassembly line balancing with multi-robotic workstations," *European Journal of Operational Research*, vol. 276, no. 1, pp. 160–174, 2019.

[12] C. M. Fonseca, C. A. Antunes, R. Lacour, K. Miettinen, P. M. Reed, and T. Tusar, "Visualization in multiobjective optimization," in *Understanding Complexity in Multiobjective Optimization, Report from Dagstuhl Seminar 15031*, S. Greco, K. Klamroth, J. Knowles, and G. Rudolph, Eds., Dagstuhl, 2015, pp. 129–139.

[13] J. Gettinger, E. Kiesling, C. Stummer, and R. Vetschera, "A comparison of representations for discrete multi-criteria decision problems," *Decision Support Systems*, vol. 54, no. 2, pp. 976–985, 2013.

[14] A. Haara, J. Pykäläinen, A. Tolvanen, and M. Kurttila, "Use of interactive data visualization in multi-objective forest planning," *Journal of Environmental Management*, vol. 210, pp. 71–86, 2018.

[15] C. R. Harris, K. J. Millman, S. J. van der Walt, *et al.*, "Array programming with NumPy," *Nature*, vol. 585, no. 7825, pp. 357–362, Sep. 2020.

[16] J. Heinrich, Y. Luo, A. E. Kirkpatrick, and D. Weiskopf, "Evaluation of a bundling technique for parallel coordinates," in *GRAPP & IVAPP 2012: Proceedings of the International Conference on Computer Graphics Theory and Applications and International Conference on Information Visualization Theory and Applications*, P. Richard, M. Kraus, R. S. Laramee, and J. Braz, Eds., SciTePress, 2012, pp. 594–602.

[17] J. Heinrich and D. Weiskopf, "State of the art of parallel coordinates.," *Eurographics (State of the Art Reports)*, pp. 95–116, 2013.


FIGURE 15: A GUI application to analyze datasets using SCORE bands. The GUI provides support for uploading datasets, allows the analyst to customize the visualization parameters, displays the SCORE bands as well as supporting visualizations, and allows the analyst/DM to export the visualization as an image file.

[18] P. Hekkert, "Design aesthetics: Principles of pleasure in design," *Psychology Science*, vol. 48, no. 2, pp. 157–172, 2006.

[19] J. Hettenhausen, A. Lewis, and S. Mostaghim, "Interactive multi-objective particle swarm optimization with heatmap-visualization-based user interface," *Engineering Optimization*, vol. 42, no. 2, pp. 119–139, 2010.

[20] S. Hossain, "Visualization of bioinformatics data with dash bio," in *Proceedings of the 18th Python in Science Conference*, C. Calloway, D. Lippa, D. Niederhut, and D. Shupe, Eds., 2019, pp. 126–133.

[21] H. Huang and S. Siraj, "Quantifying and reducing the complexity of multi-line charts as a visual aid in multi-criteria decision-making," *Annals of Operations Research*, to appear. DOI: 10 . 1007 / s10479 - 024 - 06090-6.

[22] A. Ibrahim, S. Rahnamayan, M. V. Martin, and K. Deb, "3D-RadVis: Visualization of Pareto front in many-objective optimization," in *2016 IEEE Congress on Evolutionary Computation (CEC)*, IEEE, 2016, pp. 736–745.

[23] A. Inselberg and B. Dimsdale, "Parallel coordinates for visualizing multidimensional geometry," in *Computer Graphics 1987, Proceedings of Computer Graphics International*, T. Kunii, Ed., Springer, 1987, pp. 25–44.

[24] P. J. Korhonen and J. Wallenius, "Visualization in the multiple objective decision-making framework," in *Multiobjective Optimization: Interactive and Evolutionary Approaches*, J. Branke, K. Deb, K. Miettinen, and R. Slowinski, Eds., Berlin: Springer, 2008, pp. 195–212.

[25] P. J. Korhonen and J. Laakso, "A visual interactive method for solving the multiple criteria problem," *European Journal of Operational Research*, vol. 24, no. 2, pp. 277–287, 1986.

[26] P. J. Korhonen and J. Laakso, "Solving generalized goal programming problems using a visual interactive approach," *European Journal of Operational Research*, vol. 26, no. 3, pp. 355–363, 1986.

[27] J. Liu, T. Dwyer, K. Marriott, J. Millar, and A. Haworth, "Understanding the relationship between interactive optimisation and visual analytics in the context of prostate brachytherapy," *IEEE Transactions on Visualization and Computer Graphics*, vol. 24, no. 1, pp. 319–329, 2018.

[28] A. V. Lotov, V. A. Bushenkov, and G. K. Kamenev, *Interactive Decision Maps: Approximation and Visual-*

*ization of Pareto Frontier*. Boston: Kluwer Academic Publishers, 2004.

[29] A. V. Lotov and K. Miettinen, "Visualizing the Pareto frontier," in *Multiobjective Optimization: Interactive and Evolutionary Approaches*, J. Branke, K. Deb, K. Miettinen, and R. Slowinski, Eds., Berlin: Springer, 2008, pp. 213–243.

[30] A. Mazumdar, T. Chugh, J. Hakanen, and K. Miettinen, "An interactive framework for offline data-driven multiobjective optimization," in *Bioinspired Optimization Methods and Their Applications*, B. Filipič, E. Minisci, and M. Vasile, Eds., Cham: Springer, 2020, pp. 97–109.

[31] K. T. McDonnell and K. Mueller, "Illustrative parallel coordinates," in *Computer Graphics Forum*, Wiley Online Library, vol. 27, 2008, pp. 1031–1038.

[32] D. Meignan, J.-M. Frayret, and G. Pesant, "Interactive planning system for forest road location," *Journal of Heuristics*, vol. 21, no. 6, pp. 789–817, 2015.

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

[34] G. Misitano, B. S. Saini, B. Afsar, B. Shavazipour, and K. Miettinen, "DESDEO: The modular and open source framework for interactive multiobjective optimization," *IEEE Access*, vol. 9, pp. 148 277–148 295, 2021.

[35] G. Palmas, M. Bachynskyi, A. Oulasvirta, H. P. Seidel, and T. Weinkauf, "An edge-bundling layout for interactive parallel coordinates," in *2014 IEEE Pacific Visualization Symposium*, IEEE, 2014, pp. 57–64.

[36] F. Pedregosa, G. Varoquaux, A. Gramfort, *et al.*, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[37] Plotly Technologies Inc. "Collaborative Data Science." (2015), [Online]. Available: https://plot.ly (visited on 01/11/2024).

[38] W. Raseman, J. Jacobson, and J. Kasprzyk, "Parasol: An open source, interactive parallel coordinates library for multi-objective decision making," *Environmental Modelling & Software*, vol. 116, pp. 153–163, 2019.

[39] J. Reback, W. McKinney, jbrockmendel, *et al.*, *Pandas-dev/pandas: Pandas*, version latest, 2020. DOI: 10.5281/zenodo.3509134.

[40] R. A. Shah, P. M. Reed, and T. W. Simpson, "Many-objective evolutionary optimisation and visual analytics for product family design," in *Multi-objective Evolutionary Optimisation for Product Design and Manufacturing*, L. Wang, A. H. C. Ng, and K. Deb, Eds. London: Springer, 2011, pp. 137–159.

[41] D. Shintyakov. "tsp-solver." (2020), [Online]. Available: https://github.com/dmishin/tsp-solver (visited on 01/11/2024).

[42] H. Smedberg and S. Bandaru, "Interactive knowledge discovery and knowledge visualization for decision support in multi-objective optimization," *European Journal of Operational Research*, vol. 306, no. 3, pp. 1311–1329, 2023.

[43] A. Talukder and K. Deb, "Paletteviz: A visualization method for functional understanding of high-dimensional Pareto-optimal data-sets to aid multi-criteria decision making," *IEEE Computational Intelligence Magazine*, vol. 15, no. 2, pp. 36–48, 2020.

[44] H. L. Trinkaus and T. Hanne, "knowCube: A visual and interactive support for multicriteria decision making," *Computers and Operations Research*, vol. 32, no. 5, pp. 1289–1309, 2005.

[45] T. Tusar and B. Filipic, "Visualization of Pareto front approximations in evolutionary multiobjective optimization: A critical review and the prosection method," *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 2, pp. 225–245, 2015.

[46] P. Virtanen, R. Gommers, T. E. Oliphant, *et al.*, "SciPy 1.0: Fundamental algorithms for scientific computing in Python," *Nature Methods*, vol. 17, pp. 261–272, 2020.

[47] D. J. Walker, R. M. Everson, and J. E. Fieldsend, "Visualizing mutually nondominating solution sets in many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 17, no. 2, pp. 165–184, 2013.

[48] C. Ware, *Information Visualization: Perception for Design*. San Diego: Morgan Kaufmann, 2013.

[49] E. J. Wegman, "Hyper-dimensional data analysis using parallel coordinates," *Journal of the American Statistical Association*, vol. 85, pp. 664–675, 1990.

[50] M. J. Woodruff, P. M. Reed, and T. W. Simpson, "Many objective visual analytics: Rethinking the design of complex engineered systems," *Structural and Multidisciplinary Optimization*, vol. 48, no. 1, pp. 201–219, 2013.

[51] D. Yang, D. Di Stefano, M. Turrin, S. Sariyildiz, and Y. Sun, "Dynamic and interactive re-formulation of multi-objective optimization problems for conceptual architectural design exploration," *Automation in Construction*, vol. 118, article 103251, 2020.

[52] L. Zhen, M. Li, R. Cheng, D. Peng, and X. Yao, "Adjusting parallel coordinates for investigating multi-objective search," in *Simulated Evolution and Learning*, Y. Shi, K. Tan, M. Zhang, *et al.*, Eds., Cham: Springer, 2017, pp. 224–235.

[53] H. Zhou, W. Cui, H. Qu, Y. Wu, X. Yuan, and W. Zhuo, "Splatting the lines in parallel coordinates," in *Computer Graphics Forum*, Wiley Online Library, vol. 28, 2009, pp. 759–766.

[54] H. Zhou, X. Yuan, H. Qu, W. Cui, and B. Chen, "Visual clustering in parallel coordinates," in *Computer Graphics Forum*, Wiley Online Library, vol. 27, 2008, pp. 1047–1054.

**B. S. SAINI** is a Postdoctoral Researcher in the Multiobjective Optimization Group at the University of Jyväskylä. He received his M.Tech degree from the Indian Institute of Technology Kharagpur in 2018 and his PhD from the University of Jyväskylä in 2022. He was awarded the MCDM Doctoral Dissertation award in 2024 by the International Society on Multiple Criteria Decision Making. His research interests include multiobjective optimization, data visualization, data-driven optimization, and development of evolutionary algorithms. He is also one of the primary contributors to the DESDEO framework.

**K. MIETTINEN** received her Ph.D. degree in mathematical information technology from the University of Jyväskylä (JYU), Finland. She is the vice-rector of the JYU as well as a Professor of industrial optimization. She heads the Research Group on Multiobjective Optimization and is the Director of the thematic research area Decision Analytics Utilizing Causal Models and Multiobjective Optimization (DEMO, jyu.fi/demo). With her group, she develops the open source software framework DESDEO for interactive multiobjective optimization methods (desdeo.it.jyu.fi). She has authored over 220 refereed journal, proceedings, and collection articles; edited 20 proceedings, collections, and special issues; and written a monograph on nonlinear multiobjective optimization. Her research interests include theory, methods, applications, software and visualizations of nonlinear multiobjective optimization. She is a member of the Finnish Academy of Science and Letters, Section of Science and has received the Georg Cantor Award of the International Society on Multiple Criteria Decision Making.

**K. KLAMROTH** is a Professor for Mathematical Optimization at the University of Wuppertal in Germany. She received her PhD from the University of Braunschweig in 1994, and her Habilitation degree from the University of Kaiserslautern in 2002. She has a research focus on multiobjective optimization, spanning the bridge from modelling and theoretical analysis to decision making tools and algorithm development and testing, and covering discrete and continuous problems.

**R. E. STEUER** is the Sanford Family Chair of Business at the University of Georgia. His undergraduate, masters, and PhD are from Brown, Cornell, and the University of North Carolina. He is the author of Multiple Criteria Optimization: Theory, Computation and Application and over 110 research articles. Dr. Steuer's research interests are in multiple-attribute portfolio theory, efficient sets and surfaces, and multiple criteria decision making.

**K. DÄCHERT** holds a PhD in mathematics from the University of Wuppertal, Germany. Since 2021 she is associate professor for Mathematics/Operations Research at the University of Applied Sciences of Dresden, Germany. Her research interests include the applications of operations research problems, in general, and the development of algorithms for multi-objective optimization problems, in particular.