# Connections of Reference Vectors and Different Types of Preference Information in Interactive Multiobjective Evolutionary Algorithms

Jussi Hakanen, Tinkle Chugh, Karthik Sindhya, Yaochu Jin, Kaisa Miettinen
University of Jyvaskyla, Department of Mathematical Information Technology,
P.O. Box 35 (Agora), FI-40014 University of Jyvaskyla, Finland
Email: jussi.hakanen@jyu.fi

***Abstract*—We study how different types of preference information coming from a human decision maker can be utilized in an interactive multiobjective evolutionary optimization algorithm (MOEA). The idea is to convert different types of preference information into a unified format which can then be utilized in an interactive MOEA to guide the search towards the most preferred solution(s). The format chosen here is a set of reference vectors which is used within the interactive version of the reference vector guided evolutionary algorithm (RVEA). The proposed interactive RVEA is then applied to the multiple-disk clutch brake design problem with five objectives to demonstrate the potential of the idea in supporting decision making in optimization problems involving more than three objectives.**

## I. INTRODUCTION

In real-world optimization problems, there typically are multiple conflicting objectives involved and it is often required to come up with a single solution to be implemented in practice and, therefore, identifying the most preferred solution is of high importance. In multiple criteria decision making (MCDM), decision maker's (DM) preferences have been used to identify the most preferred solution between multiple conflicting objectives for several decades (see e.g., [1], [2]). The main reason for using preferences is that Pareto optimal solutions in multiobjective optimization are incomparable without some additional information.

In the field of evolutionary multiobjective optimization (EMO), a traditional approach has been to approximate the whole Pareto front containing estimates of all Pareto optimal solutions. During the last 15+ years in EMO, DM's preferences have also been included to multiobjective evolutionary algorithms (MOEAs) to avoid the need of approximating the whole Pareto front which can be a challenging task especially when the number of objective functions is higher than three (see e.g. a recent review of preferences in MOEAs in [3]). By utilizing preferences to guide the search, the algorithm is able to approximate a sub region of the Pareto front (sometimes referred to as a region of interest) or even to identify a single most preferred solution as is typically the case in the MCDM approaches as mentioned above.

There exist different ways to utilize preferences in multiobjective optimization [1]: before optimization (a priori), after optimization (a posteriori) or iteratively during optimization (interactive approach). Interactive approaches have several benefits as identified in [1], [4] and, therefore, can be seen as a promising approach for real-world problems. In addition to a priori and a posteriori ways of including preferences, utilizing DM's preferences interactively has become more popular in EMO during the last 10 years and a number of interactive MOEAs have been developed [3]. Although different types of preference information have been used within interactive MOEAs (e.g. reference points, pairwise comparisons, weights, preferred ranges for objectives, selecting preferred solutions, objective ranking), existing algorithms can consider only one type (or few) of preference information. In the MCDM field, it has been found that different types of preference information are suited for different DMs/problems and that it can be beneficial to be able to change the type of preferences during the search process [5], [6], [7].

The contribution of this paper is two-fold: First, we show how to unify various types of DM's preferences. Here, unifying means that different types of preference information are converted into a single format that can then be used within an interactive MOEA to guide the search. The format chosen here is a set of reference vectors because recent MOEAs like NSGA-III [8] and reference vector guided evolutionary algorithm (RVEA) [9] have been found promising for problems with a high number of objectives where the benefits of interactive approach are most usefull. The types of preference information considered here include selecting preferred solution(s) or not preferred solution(s), specifying a reference point consisting of desired values for each objective, and specifying preferred ranges for each objective. As the second part of the contribution, we demonstrate the potential of utilization of different types of preference information in decision making for problems with a high number of objectives. In practice, we introduce an interactive version of RVEA which offers a convenient way for the DM to iteratively guide the search for the most preferred solution. As a facilitator for interaction, we propose a visually driven user interface that uses parallel coordinate plots (PCPs) to both visualize obtained solutions and a way for the DM to express preferences.

The rest of the paper is organized as follows. In Section II, a brief overview of characteristics of interactive MOEAs is provided while the unification of preference information is presented in Section III. To demonstrate the ideas in practice, interactive RVEA is introduced in Section IV along with desired properties for a visually driven user interface. In Section V, interactive RVEA is applied to a multiple-disk clutch brake design problem having five objectives and conclusions and future research ideas are given in Section VI.

## II. Interactive MOEAs and preference information

Several interactive multiobjective evolutionary algorithms have been proposed in the literature during the last 10 years. A recent review of MOEAs incorporating preference information can be found in [3] where also a number of interactive algorithms have been reviewed. The main idea in these algorithms is to combine benefits of MOEAs and interactive MCDM approaches to identify the most preferred solution. In other words, to combine a population based search procedure and interactive preference elicitation. Interactive MOEAs have a number of characteristics that distinguish them from each other in the following aspects: 1) what is the type of preference information asked from the DM, 2) how the preference information is used in the algorithm, 3) what is the stopping criterion, and 4) what is the output of the algorithm.

Several different types of preference information are considered in interactive MOEAs and the most widely used types are reference points [10], [11], [12], [13], [14], [15], [16], [17], pairwise comparisons ([18], [19], [20], [21], [22], [23], [24]) and selecting preferred solutions [14], [25], [26], [27], [28], [29]. However, most of the proposed approaches utilize a fixed type of preference information, although it has been shown in [5], [6], [7] that allowing the DM to express preferences in different ways is beneficial. The exceptions among interactive MOEAs are [10], [11], [16], [17]. In [10], [11], a collection of different modules was introduced to enable decision making after running a normal MOEA. Different types of preference information are included via different scalarizations but they do not operate within a single algorithm. The preference-based interactive evolutionary algorithm [16] uses both reference points and the weights of the achievement scalarizing function as preferences while the interactive preference-inspired co-evolutionary algorithm [17] enables a DM to express preferences either as a reference point or as weights for the objectives. To summarize, in the current interactive MOEAs either the ways of expressing preferences are limited or they have been integrated into the algorithm as separate modules.

The preference information is used within interactive MOEAs in different ways. For instance, preferences can be used to improve solutions found after termination of MOEA [10], [11], to estimate a utility function [22], [24], to modify the domination principle [12], [19], [20], [25], [28], [29], [30], to adjust a distribution of weights [14], [15], [26] or to combine a fitness function and a scalarization [13].

There are mainly three kinds of stopping criteria in interactive MOEAs. The first one is based on a fixed budget of function evaluations/generations [23], [27], [31] or DM interactions [29], [30], [32], the second is based on expected improvement of solutions [20], [28] and the last one lets a DM stop when (s)he is satisfied with the solution obtained [10], [11], [13], [14], [15], [16], [21], [22], [24], [25], [27], [31].

Although MOEAs are population based algorithms, the output of interactive MOEAs is typically a single preferred solution [10], [11], [13], [14], [15], [16], [18], [20], [22], [24], [25], [27], [28], [29], [30]. However, output of some algorithms is an approximation of the region of interest [12], [19], [23], [26], [31].

To summarize, there does not exist an interactive MOEA that can use different types of preferences. To address this issue, in this paper, we introduce interactive RVEA that is able to convert four different forms of preferences into one single format and then uses this for guiding the search towards preferred solution(s). In addition, it will use the preferences in modifying the set of reference vectors, terminate when the DM wants, and output one preferred optimal solution.

## III. Converting preference information into reference vectors

As mentioned, interactive MOEAs have typically requested preferences from the DM in a predefined format. In this paper, our aim is to offer a DM different ways of expressing preferences and treat them inside the algorithm in a unified way. Due to promising reference vector based approaches for problems with a high number of objectives (e.g. NSGA-III [8] and RVEA [9]), we describe how different types of preferences can be converted into an arrangement of a set of reference vectors. The types of preferences considered are selecting preferred (or not preferred) solution(s) from a given set of solutions, specifying a reference point and specifying preferred ranges for the objectives. Note that specifying desired values or ranges for objectives has been found a cognitively valid way of expressing preferences in [33]. Here, we do not include weights as a way of expressing preferences since determining weights is not a cognitively valid approach [33] and the relation between the given weights and the obtained solutions can be confusing for the DM (see, e.g. [34]). Furthermore, we also do not include pairwise comprisons here since we do not assume the DM to have any value function because (s)he is free to change his/her mind during the solution process when new information is obtained.

### A. Selecting preferred solution

The first way of expressing preferences is to select a preferred solution among a set of solutions. Let us denote a set of uniformly distributed reference vectors by $V = \{\boldsymbol{v}^i \in \mathbb{R}^k \mid i = 1, \dots, m\}$ and assume that a DM selects a solution $\boldsymbol{z} \in \mathbb{R}^k$. Then, each reference vector $\boldsymbol{v}^i \in R$ can be rearranged as [9]

$$\bar{\boldsymbol{v}}^i = \frac{r \cdot \boldsymbol{v}^i + (1-r) \cdot \boldsymbol{v}^c}{||r \cdot \boldsymbol{v}^i + (1-r) \cdot \boldsymbol{v}^c||}, \quad (1)$$

where $\boldsymbol{v}^c = \boldsymbol{z}/||\boldsymbol{z}||$ and $r \in (0,1)$. The parameter $r$ determines how close the reference vectors are to the central vector $\boldsymbol{v}^c$ defined by using the selected solution $\boldsymbol{z}$. If $r$ is close to zero, the vectors will be close to $\boldsymbol{v}^c$ and, if it is close to one, the vectors will not change much. If the DM wants to select multiple solutions, then the same procedure can be used by repeating it for each selected solution. In that case, the reference vectors can be equally distributed for the selected solutions or the number of reference vectors can be increased if more reference vectors are needed for each selected solution.

### B. Specifying non-preferred solutions

An alternative way of expressing preferences is identifying unacceptable solutions. Again, let $V$ denote a set of uniformly distributed reference vectors and assume that the DM indicates that a solution $\boldsymbol{z} \in \mathbb{R}^k$ is not preferred. Based on that information, the Euclidean distance between $\boldsymbol{v}^c = \boldsymbol{z}/||\boldsymbol{z}||$ and each of the reference vectors $\boldsymbol{v}^i \in \mathbb{R}^k$ can be calculated and those reference vectors that are closer than a predefined distance can be either removed or re-positioned somewhere else. Further, the DM can also indicate several non-preferred solutions and, in that case, the above procedure needs to be repeated for each of them. The techniques for modifying the set of reference vectors based on preferred and non-preferred solutions described above are just examples how it can be done. For example, it is also possible to remove reference vectors far from a preferred solution or move reference vectors further away from non-preferred solutions.

### C. Specifying a reference point

One of the most widely used ways of expressing preferences is to specify desired values for each of the objective functions. Those values then form a reference point $\bar{\boldsymbol{z}} \in \mathbb{R}^k$ describing the preferences. The reference point can then be used to re-position a set of uniformly distributed reference vectors $V$ by choosing $\boldsymbol{v}^c = \bar{\boldsymbol{z}}/||\bar{\boldsymbol{z}}||$ and using (1) for each of the reference vectors $\boldsymbol{v}^i$.

### D. Specifying preferred ranges

The preferred values for the objectives can be in some range instead of a fixed value. In that case, the preference information is in the form of a preferred range $[f_i^l, f_i^u]$ for each objective function $f_i$. This results with a $k$-dimensional hyperbox $[f_1^l, f_1^u] \times \cdots \times [f_k^l, f_k^u]$ in the objective space. One way of re-positioning the reference vectors to correspond to the hyperbox is to use Latin hypercube sampling in the hyperbox to get vectors $\boldsymbol{w}^i$, $i = 1, \ldots, m$, and, further, set $\boldsymbol{v}^i = \boldsymbol{w}^i/||\boldsymbol{w}^i||$. If the size of the specified ranges is very small, then the midpoint of the ranges can be considered as a reference point and reference vectors updated accordingly.

## IV. Example interactive MOEA

To illustrate our idea, we present here an interactive MOEA based on the reference vector guided evolutionary algorithm RVEA [9]. The RVEA algorithm decomposes the objective space into subspaces using reference vectors and balances between convergence and diversity with an angle penalized distance scalarization. More details about RVEA can be found in [9]. It has been shown in [9] that RVEA can be used to take into account a priori preferences by adjusting the set of reference vectors accordingly. That forms the basis for our interactive RVEA algorithm which is able to consider different types of preference information as described in Section III.

### A. Interactive RVEA

The main steps of interactive RVEA are shown in Algorithm 1. As input, the algorithm requires three parameters: the number of reference vectors used, the number of generations to run RVEA between the interactions, and the number of solutions to be shown to the DM at each interaction. The number of reference vectors $N_V$ used should increase with the number of objective functions but, on the other hand, it should be much smaller than without the use of preferences [8], [9]. For example in [9], ten reference vectors were used to find preferred solutions for three-objective problems while in [8], only five and ten reference points were used for three- and 10-objective problems to find preferred solutions, respectively. Note that in both [8] and [9] with a priori preferences, it is mentioned that the extreme reference points/vectors (i.e. vectors with one component equal to 1 and the rest equal to 0) should be added in order to appropriately normalize and translate the objectives, respectively. Therefore, we also always add the extreme reference vectors in interactive RVEA.

**Algorithm 1** Interactive RVEA

**Input:** $N_V$ = number of reference vectors; $N_G$ = a number of generations to run RVEA; $N_S$ = a number of solutions to be shown to DM at each interaction
**Output:** The solution most preferred by DM
1: Create a set of uniformly distributed unit reference vectors $V_0$ of size $N_V$, an initial population $P_0$ of size $N_V$ randomly, and set interaction counter *it* = 0
2: Run RVEA for $N_G$ generations with initial population $P_{it}$ and reference vector set $V_{it}$
3: Show $N_S$ solutions of the final population of RVEA to DM
4: If DM wants to stop, go to step 9
5: Ask DM to indicate preferences
6: Adjust $V_{it+1}$ based on preferences
7: Set $P_{it+1}$ as the final population of RVEA from step 3
8: Update $it = it + 1$ and go to step 2
9: Ask DM to indicate the most preferred solution as the final solution

The second input parameter, i.e., the number of generations $N_G$ to run RVEA affects how fast the solutions converge towards the Pareto front. In interactive RVEA, it is not required that the solutions are necessarily close to the Pareto front in the beginning of the solution process, so the number of generations used can be set e.g. according to the available budget for function evaluations. The third input parameter, i.e., the number of solutions $N_S$ shown is determined by the DM and must be between one and $N_V$.

In the first step, uniformly distributed unit reference vectors and a random initial population are generated to run the standard RVEA for $N_G$ generations in step 2 before the first interaction with the DM. Then in step 3, $N_S$ solutions from the final population obtained from step 2 are presented to the DM. If the size of the final population is greater than $N_S$, then clustering is used to obtain $N_S$ solutions to be shown to the DM. Step 4 is the termination step for the algorithm where the DM has to choose whether to continue searching for better solutions or to stop the search process and identify the most preferred solution among the ones shown in step 9. Convergence based on termination by the DM is often referred to as psychological convergence [4]. If the DM wants to continue, (s)he is assumed to provide new preferences in step 5 in some of the forms described in Section III. Based on the new preferences, the set of reference vectors is adjusted accordingly in step 6. Note that finally the extreme reference vectors are added to $V_{it+1}$. For the next run of RVEA, the final population from the previous run is used as the initial population. Finally, interaction counter $it$ is updated and the algorithm is continued from step 2.

### B. Visually driven user interface

An important part of the implementation of an interactive multiobjective optimization method is a user interface which facilitates interaction with the DM. A user interface is used for visualizing the solutions to the DM, the DM evaluating those solutions and the DM providing preferences for the subsequent iteration. Essential to these tasks is the capability of visualizing high dimensional data and visual interaction (meaning that visualization techniques are dynamic instead of static and they respond to the DM's actions). Typically in the literature, interaction between the DM and the algorithm is considered only in the algorithmic level and no effort is devoted to an actual implementation of the interaction. Next, we briefly summarize the few ideas presented to implement the interaction.

In [27], a heatmap based visualization was combined with numerical values of the solutions and a visualization of a solution in the decision space (2D shape). In [32], a parallel coordinate plot was combined with 2D scatter plots and a visualization of solutions in the decision space (2D/3D shape). The parallel coordinate plot and 2D scatter plot were linked, i.e., when a solution is highlighted in one of the plots, it will also be highlighted in the other one to enable uncovering hidden dependencies [35], [36]. This approach was further improved in [37] where usage of multiple 2D scatter plots and free positioning of the plots was enabled. Note that in all the three approaches, the objective function and the decision variable values are shown in the same visualization. Sketches of dynamic sliders for providing the preferred ranges for the objectives was proposed in [31] where the dynamic nature gives the DM a feeling of a responsive system. Parallel coordinate plots were linked with 3D scatter plots in [38] while [17] proposed interaction without any numerical values by brushing interesting regions in the objective space although not many details were given. Finally, visual steering was proposed in [39] utilizing the ATSV visualization toolkit. (ATSV provides different types of visualization techniques including interactive 3D plots and supports linking.)

Our goal is to develop a visually driven user interface which is responsive and fully supports the interaction with the DM. We identify the required properties of such an interface and demonstrate some of them together with the interactive RVEA. The earlier approaches described above provide good ideas for developing our interface. First, we start from visualization capabilities of high dimensional data and for that we use parallel coordinate plots. For example, widely used scatter plots can not be directly used to visualize more than three dimensional data although more dimensions can be added e.g. by using color, size or orientation of the markers. On the other hand, widely used PCPs scale quite well for high dimensions and their usage for visualizing high dimensional data has been encouraged e.g. in [38], [35], [40]. Secondly, the interface should be responsive as already mentioned and one way to support that is to use dynamic filtering (also known as brushing) as in [31]. In other words, when the DM adjusts the filters, the system dynamically shows only those solutions that remain within the filters. This is usefull e.g. when several solutions are to be examined simultaneously. Another way of increasing responsiveness is to use linked plots which is highly encouraged in the visualization literature [35], [36]. In our case, it can mean linking PCPs with e.g. 3D scatter plots as was done in [38]. In the numerical example of this paper, we use the web based PCP tool developed in Prof. Patrick Reed's group in Cornell [41] which also offers responsive brushing. An example of visualization in our user interface is shown in Figure 1. Adding linked plots to it will be a future research topic.

## V. Numerical example

To demonstrate the capabilities of the interactive RVEA algorithm, we next apply it to a multiple-disk clutch brake design problem introduced in [42]. We chose this because it describes a practical problem where the objective functions have a real meaning when compared e.g. to the various test problems typically used to test MOEAs. This makes the interactive solution process meaningful since the preferences and actions of the DM have some justification.

### A. Multiple-disk clutch brake design

The problem describes optimal design of multiple-disk clutch brake and has five objective functions, five decision variables and eight inequality constraints. The objectives are to minimize 1) mass of the brake $[kg]$, 2) stopping time $[s]$, 3) number of friction surfaces, 4) outer radius $[mm]$, and 5) actuating force $[N]$. Due to different scales of the objectives, they are normalized inside the algorithm based on the current population. The decision variables are 1) the inner radius $[mm]$, 2) the outer radius $[mm]$, 3) the thickness of discs $[mm]$, 4) the actuating force $[N]$, and 5) the number of friction surfaces. The resulting five objective optimization problem is of the form

$$\begin{aligned}
&\text{minimize} && f_1 := \text{mass of the brake [kg]},\\
&\text{minimize} && f_2 := \text{stopping time [s]},\\
&\text{minimize} && f_3 := \text{number of friction surfaces},\\
&\text{minimize} && f_4 := \text{outer radius [mm]},\\
&\text{minimize} && f_5 := \text{actuating force [N]}\\
\\
&\text{subject to} && x_1 := \text{inner radius [mm]},\\
& && x_2 := \text{outer radius [mm]},\\
& && x_3 := \text{thickness of the disc [mm]},\\
& && x_4 := \text{actuating force [N]},\\
& && x_5 := \text{number of friction surfaces},\\
& && \boldsymbol{x} = (x_1, \ldots, x_5) \in S,
\end{aligned} \tag{2}$$

where $S$ denotes the feasible decision space with respect to variable bounds $60 \leq x_1 \leq 80$, $90 \leq x_2 \leq 110$, $1 \leq x_3 \leq 3$, $600 \leq x_4 \leq 1000$ and $1 \leq x_5 \leq 10$ as well as the inequality constraints. Note that objectives $f_3 - f_5$ are also decision variables. In addition to [42], the problem has been considered in [43] (with objectives $f_1$ and $f_2$) and [44]. More details can be found in [43].

*B. Interactive solution process*

Next, we describe an interactive solution process for finding the most preferred multiple-disk clutch brake design. Typically in the EMO community, the performance of novel algorithms is measured by using various performance indicators developed for describing convergence and diversity. This is well justified when the goal is to approximate the whole Pareto front (although challenges will emerge when the number of objectives increases). Some research has also been conducted to evaluate the performance of MOEAs utilizing a priori preferences [45], [46] but evaluating the performance of interactive methods remains an under-explored research topic. Some attempts have been made towards this by using a fixed value function to simulate DM's preferences (e.g. [20], [29]). However, this does not describe well the situation with a human decision maker since one of the main advantages of interactive methods is that they allow the DM to change his/her preferences if needed when new insight is gained from the behaviour of the problem during the search process.

A preliminary approach to automatic performance evaluation of interactive methods has been presented in [47] where the idea is to mimic the behaviour of a human DM who adjusts preferences based on insight gained during the search. In this paper, we aim at illustrating the potential of interactive RVEA in decision making with a high number of objectives. Further, we do not focus on performance evaluation as such since the proposed interactive RVEA is our first attempt to realize an interactive MOEA capable of utilizing different types of preference information and it is still under development. Above, we have identified aspects that still need further research and the performance evaluation is one of our future research topics in addition to those.

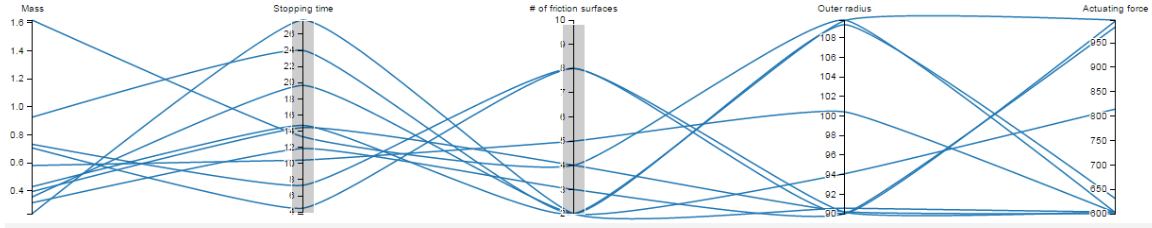
The parameter values used in this study were $r = 0.15$, $N_V = 10 + 5 = 15$ (includes the extreme reference vectors), $N_G = 1000$ (which means around 15000 function evaluations between interactions) and $N_S = 10$. These values are chosen just to demonstrate the performance and their effect on the performance is left as a future research topic due to the page limit. To reduce the number of solutions to be $N_S$, k-means clustering was used and the closest solution to the cluster centroid was chosen. For the first interaction, the solutions shown to the DM are presented in Figure 1. The DM wanted first to know what kind of solutions are available within the ranges $0.400 \leq f_1 \leq 1.00$, $6.00 \leq f_2 \leq 15.0$, $3.00 \leq f_3 \leq 6.00$, $92.0 \leq f_4 \leq 100.0$, and $650.0 \leq f_5 \leq 850.0$. Based on the ranges, the reference vectors were updated as described in Section III and the solutions obtained after running RVEA were following quite well the given ranges as can be seen in Figure 2.

Among the solutions shown, the DM liked solution $\boldsymbol{z} = (0.555, 9.61, 6.00, 90.2, 607.8)$ since it had the smallest value for the stopping time. Therefore, he selected it as a preferred one among the solutions shown and, thus, wanted to see more solutions around it. Again, the set of reference vectors was updated based on the preferred solution and new solutions were computed by running RVEA. The solutions obtained are shown in Figure 3 and, as can be observed, they were similar to the ones obtained in the previous iteration. In fact, the values for the stopping time were impaired a bit while the number of friction surfaces, outer radius and actuating force improved a little as a consequence.

This time, the DM was interested to see whether he could get the stopping time closer to 9.0 by giving a reference point $\bar{\boldsymbol{z}}^1 = (0.600, 9.50, 5.00, 92.0, 650.0)$. After running RVEA again with an updated set of reference vectors, solutions shown in Figure 4 were obtained. The solutions obtained did not have values for the stopping time closer to 9.0 as desired, so the DM provided a new reference point $\bar{\boldsymbol{z}}^2 = (0.650, 9.00, 6.00, 92.0, 700.0)$ with even a smaller value for the stopping time and a bit larger values for the number of friction surfaces and actuating force that the DM thought to be the limiting factors for improved the stopping time.

The solutions obtained corresponding to the new reference point are shown in Figure 5 and indeed there are solutions with a smaller stopping time. Now it can be observed that the three solutions with the smallest stopping times seem to correspond to the number of friction surfaces equal to 6 and, further, the stopping time reduces with an increased actuating force. The DM was satisfied with the solutions obtained and did not want to continue. Finally, the chose the solution $\boldsymbol{z}^* = (0.700, 9.27, 6.00, 90.9, 647.4)$ as the most preferred one because it had a small enough stopping time without too much increase on the other objectives.

The interactive solution process is summarized in Table I. As can be seen, the DM utilized different types of preference information to perform different tasks (investigate a region, search the surroundings of a promising solution and look for improved/missing solutions). These kinds of tasks are typical in decision making processes and performing them is enabled by the interactive RVEA.




| group | Mass | Stopping time | # of friction surfaces | Outer radius | Actuating force |
|---|---|---|---|---|---|
| Group 1 | 0.235811 | 27.742166 | 2.000000 | 90.569680 | 603.343728 |
| Group 1 | 0.705733 | 4.459578 | 8.000000 | 90.000000 | 994.438079 |
| Group 1 | 0.428336 | 14.708595 | 2.000000 | 109.741118 | 996.578927 |
| Group 1 | 0.731799 | 7.301466 | 8.000000 | 90.164304 | 603.757257 |
| Group 1 | 0.925019 | 23.984961 | 2.000000 | 109.342227 | 631.106495 |
| Group 1 | 0.580059 | 10.411890 | 5.000000 | 100.424939 | 602.340449 |
| Group 1 | 0.393966 | 14.438466 | 4.000000 | 90.207399 | 600.000000 |
| Group 1 | 0.313738 | 11.850191 | 3.000000 | 90.003211 | 982.438796 |
| Group 1 | 0.355308 | 19.655334 | 2.000000 | 94.088685 | 814.301742 |
| Group 1 | 1.618459 | 13.299762 | 4.000000 | 109.814138 | 602.407833 |

Fig. 1. A screenshot of the PCP based user interface utilizing [41]. Solutions obtained after the initial run of RVEA without preferences are shown before the first interaction. In addition to PCP of the solutions, the objective function values of the solutions are shown as numbers.

TABLE I
ACTIONS OF THE DM DURING THE INTERACTIVE SOLUTION PROCESS.

| Interaction # | DM action |
|---|---|
| **1** | Preferred ranges: $\boldsymbol{f}^l = (0.400, 6.00, 3.00, 92.0, 650.0)$, $\boldsymbol{f}^u = (1.00, 15.0, 6.00, 100.0, 850.0)$ |
| **2** | Select preferred solution: $\boldsymbol{z} = (0.555, 9.61, 6.00, 90.2, 607.8)$ |
| **3** | Reference point: $\bar{\boldsymbol{z}}^1 = (0.600, 9.50, 5.00, 92.0, 650.0)$ |
| **4** | Reference point: $\bar{\boldsymbol{z}}^2 = (0.650, 9.00, 6.00, 92.0, 700.0)$ |
| **5** | Final solution: $\boldsymbol{z}^* = (0.700, 9.27, 6.00, 90.9, 647.4)$ |

## VI. CONCLUSIONS

The aim of this paper is to improve interactive multiobjective evolutionary algorithms to further support decision making in situations with more than three objectives. We have contributed to that by presenting an approach to unify four different types of preference information and included this into the interactive reference vector guided evolutionary algorithm introduced here. In addition, we have identified some desired properties for a visually driven user interface which is an essential part of the interaction between a DM and an interactive algorithm. The potential of the interactive algorithm introduced together with the first prototype of a visually driven user interface has been demonstrated by describing an interactive solution process of a multiple-disk clutch design problem with five objectives. Based on the solution process, the possibility of using different types of preference information within a single interactive algorithm seems promising for further development. As future research topics we will focus on sensitivity analysis of the parameters in interactive RVEA, evaluating interactive MOEAs, improving the prototype of the visually driven user interface and developing a version of the algorithm for computationally expensive optimization problems.

## ACKNOWLEDGMENT

This work was supported by the FiDiPro project DeCoMo funded by TEKES, The Finnish Funding Agency for Innovation. The authors would like to acknowledge Prof. Patrick Reed's group Decision Analytics for Complex Systems (Cornell University) related to their PCP plotting tool.

## REFERENCES

[1] K. Miettinen, *Nonlinear Multiobjective Optimization*. Boston: Kluwer Academic Publishers, 1999.
[2] V. Belton and T. J. Stewart, *Multiple Criteria Decision Analysis: An Integrated Approach*. Kluwer, 2002.
[3] R. Purshouse, K. Deb, M. Mansor, S. Mostaghim, and R. Wang, "A review of hybrid evolutionary multiple criteria decision making methods," in *2014 IEEE Congress on Evolutionary Computation*. IEEE, 2014, pp. 1147–1154.
[4] K. Miettinen, J. Hakanen, and D. Podkopaev, "Interactive nonlinear multiobjective optimization methods," in *Multiple Criteria Decision Analysis: State of the Art Surveys*, 2nd ed., S. Greco et al., Ed. New York: Springer, 2016, pp. 927–976.
[5] L. R. Gardiner and R. E. Steuer, "Unified interactive multiple objective programming," *European Journal of Operational Research*, vol. 74, no. 3, pp. 391–406, 1994.
[6] M. Luque, F. Ruiz, and K. Miettinen, "Global formulation for interactive multiobjective optimization," *OR Spectrum*, vol. 33, pp. 27–48, 2011.
[7] F. Ruiz, M. Luque, and K. Miettinen, "Improving the computational efficiency in a global formulation (GLIDE) for interactive multiobjective optimization," *Annals of Operations Research*, vol. 197, no. 1, pp. 47–70, 2012.

[8] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point based non-dominated sorting approach, part i: Solving problems with box constraints," *IEEE Transactions on Evolutionary Computation,*, vol. 18, no. 4, pp. 577–601, 2014.

[9] R. Cheng, Y. Jin, M. Olhofer, and B. Sendhoff, "A reference vector guided evolutionary algorithm for many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 5, pp. 773–791, 2016.

[10] K. Deb and S. Chaudhuri, "I-EMO: An interactive evolutionary multi-objective optimization tool," in *Pattern Recognition and Machine Intelligence*, S. Pal et al., Ed. Berlin Heidelberg: Springer, 2005, pp. 690–695.

[11] ——, "I-MODE: An Interactive Multi-objective Optimization and Decision-Making Using Evolutionary Methods," in *Evolutionary Multi-Criterion Optimization*, S. Obayashi et al., Ed. Berlin Heidelberg: Springer, 2007, pp. 788–802.

[12] K. Deb, J. Sundar, U. Bhaskara Rao, and S. Chaudhuri, "Reference point based multi-objective optimization using evolutionary algorithms," *International Journal of Computational Intelligence Research*, vol. 2, no. 3, pp. 273–286, 2006.

[13] L. Thiele, K. Miettinen, P. Korhonen, and J. Molina, "A preference-based evolutionary algorithm for multi-objective optimization," *Evolutionary Computation*, vol. 17, no. 3, pp. 411–436, 2009.

[14] A. B. Ruiz, M. Luque, K. Miettinen, and R. Saborido, "An interactive evolutionary multiobjective optimization method: Interactive wasf-ga," in *EMO 2015*, A. Gaspar-Cunha et al., Ed. Springer, 2015, pp. 249–263.

[15] A. B. Ruiz, M. Luque, F. Ruiz, and R. Saborido, "A combined interactive procedure using preference-based evolutionary multiobjective optimization. Application to the efficiency improvement of the auxiliary services of power plants," *Expert Systems with Applications*, vol. 42, no. 21, pp. 7466–7482, 2015.

[16] K. Sindhya, A. B. Ruiz, and K. Miettinen, "A preference based interactive evolutionary algorithm for multi-objective optimization: PIE," in *EMO 2011*, R. H. C. Takahashi et al., Ed. Heidelberg: Springer, 2011, pp. 212–225.

[17] R. Wang, R. Purshouse, and P. J. Fleming, ""Whatever works best for you"- a new method for a priori and progressive multi-objective optimisation," in *EMO 2013*, R. C. Purshouse et al., Ed. Heidelberg: Springer, 2013, pp. 337–351.

[18] R. Battiti and A. Passerini, "Brain–computer evolutionary multiobjective optimization: A genetic algorithm adapting to the decision maker," *IEEE Transactions on Evolutionary Computation*, vol. 14, no. 5, pp. 671–687, 2010.

[19] J. Branke, S. Greco, R. Slowinski, and P. Zielniewicz, "Interactive evolutionary multiobjective optimization using robust ordinal regression," in *Evolutionary Multi-Criterion Optimization*, M. Ehrgott et al., Ed. Berlin Heidelberg: Springer, 2009, pp. 554–568.

[20] K. Deb, A. Sinha, P. Korhonen, and J. Wallenius, "An interactive evolutionary multiobjective optimization method based on progressively approximated value functions," *Evolutionary Computation, IEEE Transactions on*, vol. 14, no. 5, pp. 723–739, 2010.

[21] S. Greco, B. Matarazzo, and R. Slowinski, "Interactive evolutionary multiobjective optimization using dominance-based rough set approach," in *2010 IEEE Congress on Evolutionary Computation*, 2010, pp. 1–8.

[22] A. Jaszkiewicz, "Interactive multiobjective optimization with the pareto memetic algorithm," *Foundations of Computing and Decision Sciences*, vol. 32, no. 1, pp. 15–31, 2007.

[23] J. Krettek, J. Braun, F. Hoffmann, T. Bertram, T. Ewald, H.-G. Schubert, and H. Lausch, "Interactive evolutionary multiobjective optimization for hydraulic valve controller parameters," in *Advanced Intelligent Mechatronics, 2009. AIM 2009. IEEE/ASME International Conference on*, 2009, pp. 816 –821.

[24] S. Phelps and M. Köksalan, "An interactive evolutionary metaheuristic for multiobjective combinatorial optimization," *Management Science*, vol. 49, no. 12, pp. 1726–1738, 2003.

[25] T. Chugh, K. Sindhya, J. Hakanen, and K. Miettinen, "An interactive simple indicator-based evolutionary algorithm (I-SIBEA) for multiobjective optimization problems," in *EMO 2015*, A. Gaspar-Cunha et al., Ed. Springer, 2015, pp. 277–291.

[26] M. Gong, F. Liu, W. Zhang, L. Jiao, and Q. Zhang, "Interactive moea/d for multi-objective decision making," in *Proceedings of the 13th Annual Conference on Genetic and Evolutionary Computation*, ser. GECCO '11. New York, NY, USA: ACM, 2011, pp. 721–728.

[27] 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.

[28] A. Sinha, P. Korhonen, J. Wallenius, and K. Deb, "An interactive evolutionary multi-objective optimization method based on polyhedral cones," in *Proceedings of learning and intelligent optimization conference (LION-2010)*, C. Blum and R. Battiti, Eds. Springer – Verlag, 2010, pp. 318–332.

[29] ——, "An interactive evolutionary multi-objective optimization algorithm with a limited number of decision maker calls," *European Journal of Operational Research*, vol. 233, no. 3, pp. 674–688, 2014.

[30] A. Sinha, A. Pandey, and K. Deb, "Solving high objective problems in fixed interactions with the decision maker," in *2012 IEEE Congress on Evolutionary Computation*, 2012, pp. 1–8.

[31] T. Aittokoski and S. Tarkkanen, "User preference extraction using dynamic query sliders in conjunction with UPS-EMO algorithm," *arXiv*, 2011, arXiv:1110.5992 [cs.NE].

[32] J. Hettenhausen, A. Lewis, M. Randall, and T. Kipouros, "Interactive multi-objective particle swarm optimisation using decision space interaction," in *Proceedings of 2013 IEEE Congress on Evolutionary Computation*. IEEE, 2013, pp. 3411–3418.

[33] O. Larichev, "Cognitive validity in design of decision aiding techniques," *Journal of Multi-Criteria Decision Analysis*, vol. 1, no. 3, pp. 127–138, 1992.

[34] H. Nakayama, "Aspiration level approach to interactive multi-objective programming and its applications," in *Advances in Multicriteria Analysis*, P. Pardalos et al., Ed. Kluwer Academic Publishers, 1995, pp. 147–174.

[35] S. Liu, W. Cui, Y. Wu, and M. Liu, "A survey on information visualization: recent advances and challenges," *The Visual Computer*, vol. 30, no. 12, pp. 1373–1393, 2014.

[36] J. Kehrer and H. Hauser, "Visualization and visual analysis of multifaceted scientific data: A survey," *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 3, pp. 495–512, 2013.

[37] J. Hettenhausen, A. Lewis, and T. Kipouros, "A web-based system for visualisation-driven interactive multi-objective optimisation," in *Procedia Computer Science*. Elsevier, 2014, vol. 29, pp. 1915–1925.

[38] 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.

[39] G. Stump, S. Lego, M. Yukish, T. W. Simpson, and J. A. Donndelinger, "Visual steering commands for trade space exploration: User-guided sampling with example," *Journal of Computing and Information Science in Engineering*, vol. 9, no. 4, pp. 1–10, 2009.

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

[41] "Web-based parallel coordinate plot," visited on 11.7.2016. [Online]. Available: https://reed.cee.cornell.edu/parallel-axis-categories/parallel/

[42] A. Osyczka, *Computer Aided Multicriterion Optimization System (CAMOS): Software Package in Fortran*. International Software Publishers, 1992.

[43] K. Deb and A. Srinivasan, "Innovization: Innovative design principles through optimization," Indian Institute of Technology Kanpur, Kanpur, India, Tech. Rep. KanGAL Report Number 2005007, 2005.

[44] P. Mackin, A. Roy, and J. Wallenius, "An interactive weight space reduction procedure for nonlinear multiple objective mathematical programming," *Mathematical Programming*, vol. 127, no. 2, pp. 425–444, 2011.

[45] A. Mohammadi, M. N. Omidvar, and X. Li, "A new performance metric for user-preference based multi-objective evolutionary algorithms," in *2013 IEEE Congress on Evolutionary Computation*. IEEE, 2013, pp. 2825–2832.

[46] K. Deb, F. Siegmund, and A. H. C. Ng, "R-HV: A metric for computing hyper-volume for reference point based EMOs," in *Proceedings of SEMCCO 2014, 5th International Conference on Swarm, Evolutionary, and Memetic Computing*, K. B. Panigrahi et al., Ed. Springer International Publishing, 2015, pp. 98–110.

[47] V. Ojalehto, D. Podkopaev, and K. Miettinen, "Towards automatic testing of reference point based interactive methods," in *Parallel Problem Solving from Nature – PPSN XIV*, J. Handl et al., Ed. Springer, 2016, pp. 483–492.

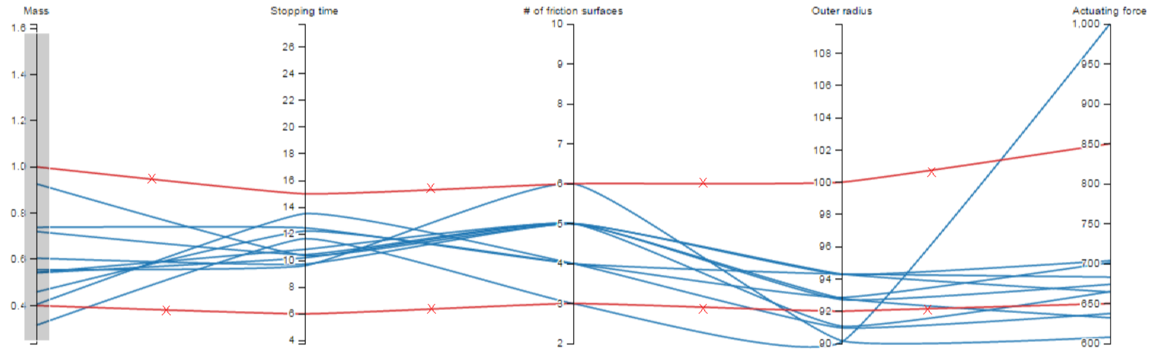



Fig. 2. Solutions shown to the DM after the first interaction including the given ranges in red (and marked with $X$).

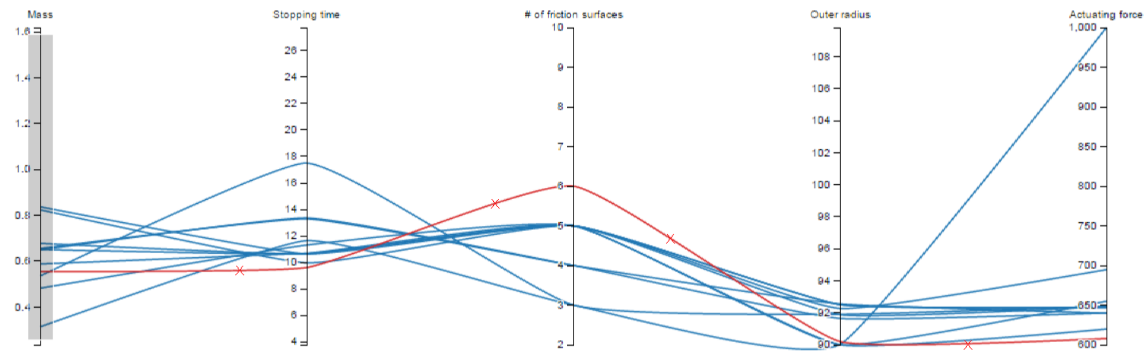



Fig. 3. Solutions shown to the DM after the second interaction including the preferred solution in red (and marked with $X$).

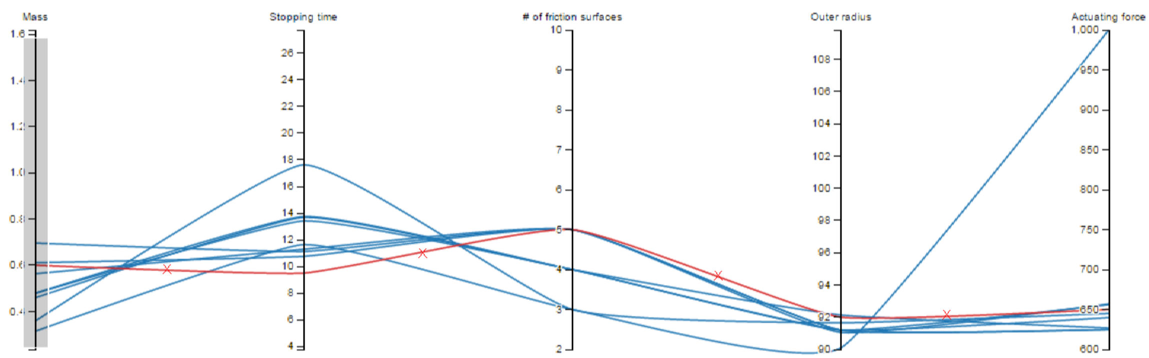



Fig. 4. Solutions shown to the DM after the third interaction including the given reference point in red (and marked with $X$).

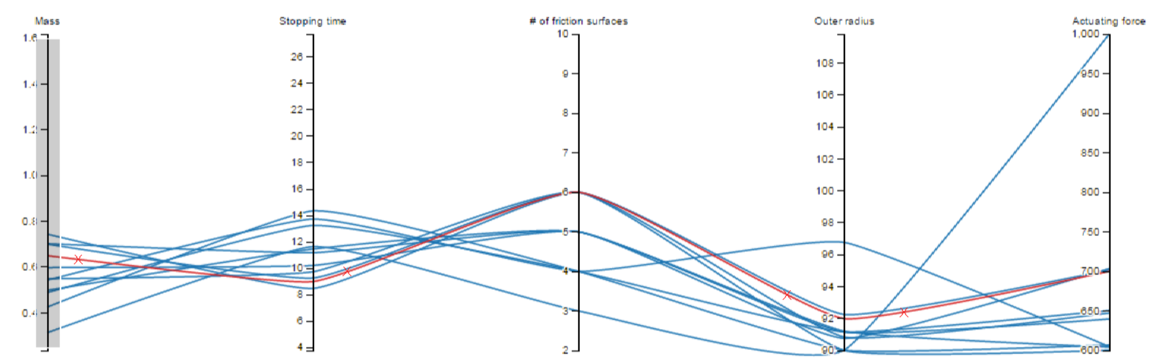



Fig. 5. Solutions shown to the DM after the fourth interaction including the given reference point in red (and marked with $X$).