Interactive multiobjective optimization of an extremely computationally expensive pump design problem

Jana Burkotová, Pouya Aghaei Pour, Tomáš Krátký & Kaisa Miettinen

Published online: 08 Sep 2023.

Submit your article to this journal ⬚

View related articles ⬚

View Crossmark data ⬚

RESEARCH ARTICLE

Interactive multiobjective optimization of an extremely computationally expensive pump design problem

Jana Burkotová [a], Pouya Aghaei Pour [b], Tomáš Krátký [c] and Kaisa Miettinen [b]

[a]Faculty of Science, Palacký University Olomouc, Olomouc, Czech Republic; [b]University of Jyvaskyla, Faculty of Information Technology, University of Jyvaskyla, Finland; [c]Centre of Hydraulic Research, Lutín, Czech Republic

ABSTRACT

The hydraulic design of a pump is a challenging optimization problem. It has multiple conflicting objective functions based on computationally very expensive (16–20 hours) numerical simulations, and simulation failures, meaning that simulation calls can be unsuccessful. In this article, a surrogate-assisted evolutionary interactive multiobjective optimization method is applied to designing a pump stator. A decision maker's preferences are iteratively incorporated into the solution process and the advantages of an interactive method are demonstrated in two areas: (1) reducing the computation time; and (2) finding a preferred solution that reflects the decision maker's preferences with a low cognitive load. The decision maker was satisfied with the interactive solution process and the final solution reflected his preferences well. Additionally, because he was familiar with the domain of the problem, the preferences he provided guided the search in directions where no failed simulations were encountered. Importantly, the applied method could save days of computation time.

1. Introduction

Real-world problems bring multiple challenges to optimization. Typically, they consist of multiple objective functions that are conflicting by nature and need to be optimized simultaneously. Therefore, no single solution, but a set of so-called Pareto optimal solutions, exists. These solutions represent different trade-offs between the objective functions, and the set of Pareto optimal solutions in the objective space is called a Pareto front.

Because evolutionary multiobjective optimization (EMO) algorithms are stochastic and population-based, they provide a set of solutions that approximates the Pareto front (Deb 2001) and can handle problems with different complexities (like non-differentiable functions, disconnected Pareto fronts, *etc.*). A drawback of EMO algorithms is that a large number of function evaluations is needed. This brings the next challenge related to real-world problems arising, for instance, in simulation-based problems: they may involve computationally expensive function evaluations — see, for example, Cai, Gao, and Li (2020), Wang *et al.* (2021) and Alizadeh, Allen, and Mistree (2020). In other words, function evaluations may need time-consuming simulations to be completed. In such cases, surrogate models (Chugh, Sindhya, *et al.* 2019) can be used to approximate the underlying functions to reduce the demand for computational resources. In this way, the expensive function evaluations can be used only when there is a need to update the accuracy of surrogate models. There

are many different ways for updating surrogate models (Jin, Wang, and Sun 2021). The process of updating surrogate models is referred to as *model management*. The last real-world optimization challenge to be mentioned here is the occurrence of failed function evaluations, which can hinder the solution process in simulation-based problems. With failed evaluations, computation time is spent but no function values are obtained. Furthermore, the possibility of such failures cannot necessarily be eliminated with constraints.

The focus of this article is on solving a (hydraulic) pump design problem (Krátký 2020), where all the above-mentioned challenges need to be considered. The aim is to maximize three conflicting objective functions that represent pump efficiencies at three different flow rates. The evaluation of objective functions involves extremely expensive computational fluid dynamics (CFD) simulations that take between 16 and 20 hours. Moreover, the geometry or mesh generation may fail for some combinations of design parameters. To solve the problem in a reasonable time, the computational resources need to be used as efficiently as possible.

In the literature, pump optimization is often approached by building surrogate(s) from the initial sampling, and the optimization is only performed on the surrogate model (Gan *et al.* 2023; Jaiswal *et al.* 2022). In contrast, this article is oriented towards methods where, in each iteration, surrogate models are updated with computationally expensive simulations. Handling infeasible designs in pump optimization has been addressed by Fracassi *et al.* (2022), where the problem was dealt with penalization and replacement of infeasible designs with the nearest feasible ones. In this article, the impact of model management strategy on the presence of infeasible designs in the solution process is also discussed.

In practice, a decision maker (DM) who is an expert in the problem domain is involved in the solution process when multiple objective functions involve trade-offs. The role of the DM is to provide preference information to identify the most preferred solution. The DM is usually interested in eventually finding one solution that is selected for implementation in practice.

Multiobjective optimization methods can be divided into three classes, depending on the role of the DM in the solution process (Miettinen 1999). Representative methods from all three classes are applied to demonstrate how a DM can participate in the solution process in them, and the positive and negative properties of the different classes of method are discussed.

The first method considered here is an example of a-posteriori methods that generate an approximation of the whole Pareto front. The K-RVEA method (Chugh *et al.* 2018) was used to solve the pump design problem by Krátký (2020). It uses Kriging-based surrogates (Sacks, Schiller, and Welch 1989) to speed up calculations. All generated solutions were presented to the DM, who selected one final solution based on his preferences.

Then the pump design problem is solved with an a-priori method, where the DM first provides hopes and expectations and the method tries to find a solution that meets the needs as well as possible. An a-priori version of K-RVEA (Chugh, Kratky, *et al.* 2019) is applied, where the DM provides preference information in the form of a *reference point* consisting of desirable values for the objective functions. This information is utilized to approximate a subregion of the Pareto front that is of interest to the DM. In this case, the DM needs to select the final solution from a more limited set than in the first case.

The final class of optimization method consists of interactive methods (Miettinen, Hakanen, and Podkopaev 2016; Miettinen, Ruiz, and Wierzbicki 2008), where the DM actively takes part in an iterative solution process, sees solution candidates reflecting his/her preferences and modifies the preferences to find the most preferred solution. Here, the recently proposed interactive K-RVEA method (Aghaei Pour *et al.* 2022) is applied. The advantage of an interactive method is that the DM can adjust preferences iteratively during the solution process when learning about the trade-offs as well as about the feasibility of his/her preferences. At the same time, computation time is saved compared to an a-posteriori method, since only solutions of interest to the DM are generated. Furthermore, the cognitive load set on the DM is limited, since the amount of information to be analysed at a time can be controlled by the DM.

The contributions of this article are twofold. First, a real-world industrial multiobjective optimization problem with extremely computationally expensive CFD simulations is successfully solved. In this case, any reduction in the computation cost is crucial. Saving a few simulator calls means saving several days of computation time. The second contribution is a comparison of different approaches to how to incorporate the DM's preferences in the solution process and a demonstration of the benefits and efficiency of an interactive method that incorporates the DM's preferences iteratively to find the most preferred solution and manages surrogate models in an efficient manner. In particular, interactive K-RVEA:

- benefits from its model management strategy, which has been shown to be important for reliable decision support of the DM;
- follows the DM's preferences and generates solutions that reflect them much better than a-priori K-RVEA, which uses different model management;
- produces comparable solutions in the DM's preferred region as a-posteriori K-RVEA with less computation time and resources;
- helps to avoid the region(s) with failed function evaluations.

The rest of this article is structured as follows. Background information, pump design problem description and a brief characterization of Kriging-assisted methods can be found in Section 2. Section 3 is devoted to results on the pump design problem obtained with non-interactive methods. Section 4 provides the main results with the interactive K-RVEA method, and demonstrates the significance of its model management strategy. The results are discussed and analysed in Section 5 and, finally, conclusions are drawn and future research directions are sketched out in Section 6.

2. Background

This section covers some basic multiobjective optimization concepts and notation needed in the rest of the article. The pump design problem is also briefly described and background information about the surrogate-assisted methods used to solve the problem is provided.

2.1. Concepts and notation

Multiobjective optimization problems considered here have the following form:

$$
\begin{aligned}
\text{maximize} \quad & f(x) = (f_1(x), \ldots, f_k(x)) \\
\text{subject to} \quad & x \in S,
\end{aligned}
\tag{1}
$$

where $f(x)$ is an objective vector in the objective space R^k. It consists of the values of the k conflicting objective functions at $x = (x_1, \ldots, x_n)^{\mathrm{T}}$, which is an n-dimensional design variable vector (design vector for short). It belongs to a feasible set S (in the design space R^n), defined by constraints. Objective vectors are referred to as *solutions*.

A solution $f(x)$ *dominates* $f(y)$ if $f_i(x) \geq f_i(y)$ for $i = 1, \ldots, k$ and $f_j(x) > f_j(y)$ for at least one index j. If $f(x)$ and $f(y)$ do not dominate each other, they are called *mutually non-dominated*. A solution is Pareto optimal if there exists no feasible solution that dominates it. As mentioned in the introduction, Pareto optimal solutions constitute a Pareto front in R^k. Note that, if a solution is on the Pareto front, it is also non-dominated, but the reverse statement does not necessarily hold. Because of the evolutionary algorithms' stochastic nature, EMO methods generate an approximation of the Pareto front, where the solutions are mutually non-dominated, but they cannot guarantee Pareto optimality.

The following two key definitions are used in this article.

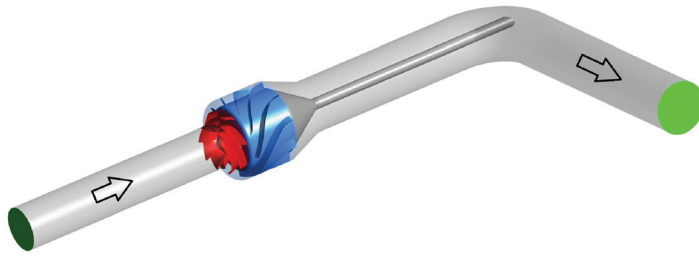- *Iteration:* one update of surrogate models is done in an iteration.

Figure 1. CFD model of the pump. The rotor (impeller) is in red and the stator part, subject to the optimization, in blue. (Colour figure online).

- *Interaction:* when a DM provides preferences, it is done in an interaction; thus, an interaction is conducted after many iterations.

2.2. The pump design problem

The optimization problem considered is related to the hydraulic design of a pump (Krátký 2020, Chapter 7). The goal is to improve the efficiency of a diagonal hydrodynamic pump. The efficiency of a pump is defined as the ratio between the energy added to the fluid and the energy spent on the pump operation. In this work, the hydraulic expert created an initial design (referred to as a *baseline design*) for the diffuser based on his experience. The performance of the design was evaluated with the ANSYS® CFX® commercial finite-volume based software. This baseline design was used as a starting point for creation of the geometry parametric model. Such an approach helps focusing the optimization search, making the optimization process less computationally demanding.

A CFD model of the problem to be solved is shown Figure 1. It consists of the following components: an inflow, an impeller (with 7 blades), a diffuser (with 11 blades) and an outflow. The inflow and outflow parts are usually prolonged to avoid any backflow occurring at the inlet and outlet. For the CFD simulations with the CFX software, first the mesh has to be created for the components. In this work, separate meshes for each component were created. For the impeller and diffuser parts, ANSYS® TurboGrid™ was used. The TurboGrid is a specialized turbomachinery tool that can create high quality structured meshes in a semi-automated way. The remaining components were meshed using ANSYS® ICEM™. Moreover, the meshes of the inflow and sealing rings were structured and of the outflow were unstructured (specifically tetra prism). The total number of nodes in the mesh of the whole design was approximately three millions. If the meshes were different between connected components in the design, the so-called *General Grid Interface* (an interpolation of values between nodes) was used in CFX. The meshed design was then connected to the CFX via an interface.

Once the meshed design had been connected to CFX, simulations were performed as fully-transient, *i.e.* with the rotating impeller, and the efficiency value was obtained as a moving average over the last two impeller rotations. Transient simulations are more computationally expensive, but also more accurate than *steady-state* simulations, as inertia forces and impeller and diffuser blades interactions are captured more realistically. However, fully-transient simulations are computationally expensive. Based on the expert's experience, 154 time steps were set for one rotation of the impeller, and for each time step, there were three inner sub-steps. This configuration leads to 2.34 degrees of impeller rotation during each time step, which is a reasonable balance between numerical simulation accuracy and computational requirements. The pump used in this work had a frequency of 995 rotations per minute, and thus the time step value was $\Delta t = \frac{60}{154 \times 995} = 0.000,391,57$ seconds. For the boundary conditions (BCs), the mass flow rate was used at the outlet and the total pressure was used at the inlet. This is common practice when doing CFD simulations for pumps, as these BC settings typically result in good numerical accuracy and stability. For turbulence modelling, the SST k–Ω model was used. Overall, one CFD simulation (of one design for one particular flow rate) took

between 16 and 20 hours on one (dual-socket) node of a high performance computing cluster. If there are enough computational resources available, it is possible to solve all the available designs and flow rates in parallel, speeding up the computations.

The objective is to improve the efficiency in the working range of the pump with an emphasis on the sub-optimal flow rates. Based on the expert's experience, only the diffuser component was the subject of optimization. The diffuser is a part of the stator and is placed downstream of the impeller. The main work of the diffuser is to guide the flow and to help the conversion of circumferential velocity (caused by the rotating impeller) to forward movement. A good design of the diffuser can improve the efficiency and performance of the whole pump. To accomplish this, three objective functions were defined: to maximize the efficiencies of a pump at 76%, 100% and 120% of the design point, where the design point is the flow rate for which the pump is designed. These three points for efficiency maximization were selected to ensure high efficiency in the whole working range of the pump. The objective function values were obtained as a result of numerical simulations, performed with ANSYS CFX software. See Figure 1 for a visualization of the CFD model. As it is not possible to have optimal flow character for different passage velocities (*i.e.* for different flow rates) with the same design, these objective functions are conflicting in nature. More details of the problem are given in Krátký (2020, Chapter 7). As the flow phenomenon is very complex, and because the blade geometry is not only decided by individual design variables, but also by the connection between the hub and shroud lines, there is no straightforward relation between the design variables and the objective functions. In general, it can be stated that, for lower flow rates, guidance of the flow is preferred (*i.e.* the stator tends to be longer and the blades more curved), while for higher flow rates, the friction losses play a more important role, and the stator geometry needs to be shorter and more 'open' (fewer curved blades).

As mentioned, the numerical simulations have a very high computation cost, combined with the need for creating a parametric geometry model and a combination of different simulation tools. It is thus mandatory to utilize the available computational resources as efficiently as possible. The solution process is also complicated by the presence of failed samples. In other words, some evaluations of the objective functions can fail, typically owing to geometry creation problems. The most typical reason for this is that, to cover a wide range of geometric shapes, hub and shroud blade angles are set independently. Owing to this, the connection between these two (hub and shroud) lines can become very skewed, and problematic for either geometry, or mesh creation. For manual geometry creation, this is not a problem, as the blades can be smoothed visually. In principle, it is possible to create a different, less error-prone, parametric model. However, such a model would be based on parameters different from the hydraulic ones that the hydraulic experts are used to — causing decision making to be more difficult. As the focus is on solving a real-world problem, based on good-practice pump design, the preference is to keep a 'standard' parametric model and to accept the existence of failed samples. Although the time required to detect whether geometry parameters induce a failure in function evaluations is usually shorter compared to CFD simulations, the presence of failed samples brings difficulties into the solution process. When simulation failures occur only rarely, the method can simply ignore these samples. However, if failures are more frequent, the process is slowed down, hindered from progress, or it may even get stuck in the infeasible region. In all cases, the allocated computational resource is left unemployed until all simulations in the current iteration are finished. Therefore, the resource is wasted while no new simulation is evaluated. This is something the optimization needs to be able to deal with.

2.3. Surrogate-assisted methods

There are many surrogate-assisted methods for multiobjective optimization in the literature, as surveyed by Chugh, Sindhya, *et al.* (2019) and Jin (2011). Among them, Kriging-based methods are popular since Kriging models provide uncertainty information on the approximation and this can be helpful in developing model management strategies (Lim *et al.* 2010).

Table 1. Comparison of the three methods in brief.

Method	Reference vectors	Infill criterion	Improvement of Kriging models
(A-posteriori) K-RVEA	Uniformly distributed	Low APD or high uncertainty	Global in whole objective space
A-priori K-RVEA	Adjusted to preferences	Low APD or high uncertainty	Local in the preferred region
Interactive K-RVEA	Adjusted to preferences	Low ASF and low uncertainty	Local close to reference vectors

K-RVEA

K-RVEA (Chugh *et al.* 2018) is a Kriging-assisted extension of the reference vector-guided evolutionary algorithm (RVEA) (Cheng *et al.* 2016). K-RVEA has been successfully applied in computationally expensive real-world optimization problems (Krátký 2020; Zhao *et al.* 2019).

K-RVEA has two main features: model management, where the method tries to balance between improving the accuracy of the surrogate models, the diversity and the convergence. After each iteration in K-RVEA, an infill criterion called angle penalized distance (APD) is applied to solutions that either selects the solution that has the highest uncertainty (has the highest potential for improving the accuracy) or selects the solution that contributes the most to the convergence and diversity of the approximated Pareto front. For more details, see Chugh *et al.* (2018). Because K-RVEA was successfully applied by Krátký (2020) to the problem considered in this article, the focus is on its variants to study how the incorporation of preference information could help in saving computation time.

A-priori K-RVEA

A-priori K-RVEA was first proposed by Chugh, Kratky, *et al.* (2019) and applied to a shape design optimization problem in a ventilation system. In this method, the DM has to provide his/her preferences as a reference point before the optimization process. Then, the procedure introduced by Hakanen *et al.* (2016) is applied to incorporate the preferences while using the Kriging models. A-priori K-RVEA uses the same model management and archive management as K-RVEA. This method was compared to K-RVEA (without preferences) by Chugh, Kratky, *et al.* (2019), and it performed significantly better than K-RVEA in the region in which the DM was interested.

Interactive K-RVEA

Interactive K-RVEA was proposed by Aghaei Pour *et al.* (2022). It was applied there in finding the optimal design of a building and shown to both reduce the computation time significantly and find solutions that followed the DM's preferences, provided as a reference point. Thus, the results were satisfactory to the DM. As the name suggests, in interactive K-RVEA, the main idea is to incorporate preferences during the optimization process and model management iteratively.

The main difference between interactive K-RVEA, a-priori K-RVEA and (a-posteriori) K-RVEA is in the model management. Interactive K-RVEA incorporates the DM's preferences when the Kriging models need to be updated. Its infill criterion uses an achievement scalarizing function (ASF) (Wierzbicki 1982) to select some of the solutions that are closest to the DM's reference point. Then, out of those solutions, the ones with a low uncertainty are selected. The idea behind this model management is to make sure that the solutions to be evaluated with the simulator have the highest chance of following the DM's preferences. An overview and a brief comparison of the three methods is provided in Table 1.

3. Results with non-interactive methods

This section is devoted to solutions generated by the non-interactive K-RVEA and a-priori K-RVEA methods for the pump design problem. With K-RVEA, the role of the DM is to choose the final solution among those generated, and with a-priori K-RVEA it is to provide a reference point and select the final solution. The DM was an expert in the hydraulic pump design domain. The goal of the DM was
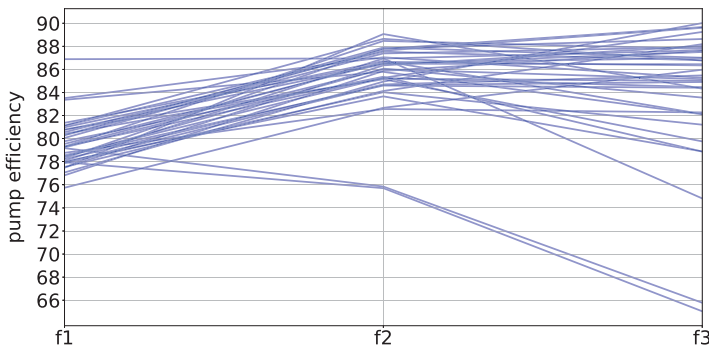
Figure 2. Objective function values of the initial population.

to outperform an existing baseline pump design. Therefore, his choices of preferences and solutions were mainly motivated by the baseline objective function values, *i.e.* $(f_1, f_2, f_3) = (81.6, 88.5, 84.5)$. In particular, the aim of the DM was to find a design without an efficiency drop at suboptimal flow rates and thus to improve the first objective function value while the other objective function values were kept at desirable levels.

The same initial dataset of 46 samples generated by Latin hypercube sampling was considered for all methods in this article. Owing to various geometry and mesh creation errors, 37 samples were successfully evaluated and 9 failed. The successful samples from the initial dataset are plotted as a parallel coordinate plot in Figure 2. In this and the following figures, solutions are plotted in blue (solid), reference points in green (dotted), final selected solutions in red (dash-dotted), and other solutions important to the DM in dark blue (dashed).

Previous results obtained by K-RVEA

First, a report on results obtained with K-RVEA in Krátký (2020) is provided. The method was set to generate three new samples in each iteration to update the Kriging models. Having multiple samples in every iteration both helped utilizing the available computational resources and minimized problems with the failed samples (as the probability of obtaining three failed samples is lower than for a single one). The failed samples were simply ignored and only the successful ones were used. In the case when all three evaluations failed, the method had to be restarted with a slightly different setting of the Kriging model. The optimization process took 45 iterations. It was terminated when the stopping criterion based on the progress in hypervolume (Zitzler and Thiele 1998) improvement was reached. During the optimization, 135 samples were generated. Out of these, 99 were successfully evaluated and the rest failed. Finally, a diverse set of solutions that approximated the Pareto front was obtained.

All 17 non-dominated solutions obtained with K-RVEA were presented to the DM. They are illustrated in Figure 3. The DM was mainly interested in finding a pump design that improved the efficiency in the suboptimal flow rate f_1 while maintaining good efficiencies in the optimal and high flow rates represented by f_2 and f_3, respectively. Therefore, the DM selected the design with a sufficiently good efficiency (*i.e.* higher than 90%) at the optimal flow rate, and with a high efficiency at 76% of the optimal flow rate. The objective function values in the final, selected, solution were

$$(87.9, \ 90.3, \ 87.6).$$

The selected solution maintained good efficiency values at the other points of the working range as well. The final solution is indicated in red (dash-dotted) in Figure 3. The DM was satisfied with it because it had all the desired properties — high efficiency in the whole working range without a drop in the lower flow rate.
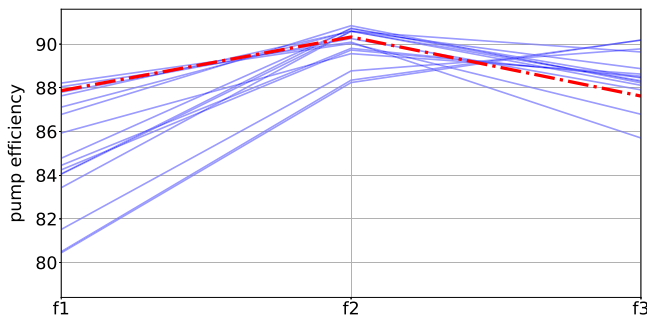
Figure 3. Non-dominated solutions generated by K-RVEA are in blue (solid). The final solution is shown in red (dash-dotted).
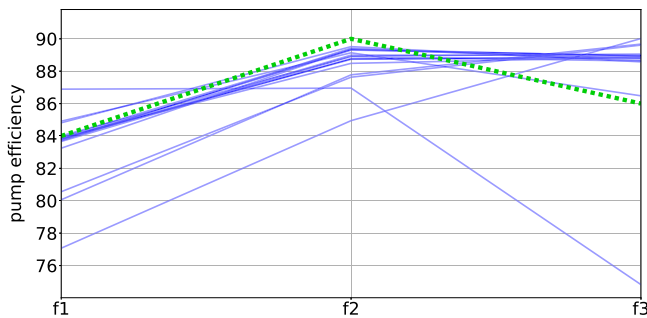

Figure 4. Non-dominated solutions generated by a-priori K-RVEA are in blue (solid). The reference point is shown in green (dotted).

Results obtained by a-priori K-RVEA

Next, the results obtained by a-priori K-RVEA are described. According to the nature of a-priori methods, the idea was to focus on the DM's preferred region in the objective space, and thus further reduce the computation cost. As mentioned, the same initial dataset as for K-RVEA was used, *i.e.* 37 samples. The DM provided preferences in the form of a reference point before the optimization. The reference point was set as

$$(84, 90, 86).$$

The components of the reference point were selected based on the DM's knowledge about the problem. The DM was expecting to obtain a solution that reached or even surpassed the reference point.

During the solution process, 111 solutions were generated in 37 iterations, 3 solutions failed and 108 were successfully evaluated. However, the DM's desires were not met, as shown in Figure 4, where 18 non-dominated solutions are depicted in blue (solid), while the reference point is depicted in green (dotted). In fact, the DM felt that a-priori K-RVEA could not follow his preferences, *i.e.* the generated solutions did not reflect his reference point. Therefore, the DM declared the solution process with a-priori K-RVEA to be unsuccessful, and could not select any final solution.

Challenges and shortcomings

As the results discussed so far have shown, solving the pump design problem is not an easy task. The K-RVEA method was able to find a diverse representation of the approximated Pareto front. As an a-posteriori method, it explored the whole objective space, and thus also produced solutions that were not of interest to the DM. Therefore, it also wasted computation time and resources. On the other hand, a-priori K-RVEA was not successful in finding satisfactory solutions at all.

4. Interactive solution process

This section is devoted to the solution process of the pump design problem using the interactive K-RVEA method. Since the DM provided preferences iteratively based on the insight learned, a description of the preferences the DM provided and what kind of solutions were generated in each interaction is given. Thus, the DM could change his preferences during the solution process according to the new information obtained about reachable solutions and their trade-offs.

As before, the same initial dataset was used (see Figure 2). The DM wanted to see 15 new solutions at each interaction. To follow the same setup as in K-RVEA where models were updated after each 20 generations with 3 new solutions, 5 iterations between each interaction were used. In this way, 15 new solutions were generated and evaluated with the simulator. The rest of the parameters of the method were set based on Aghaei Pour *et al.* (2022). Owing to the extremely computationally expensive simulations, the DM needed to wait up to four days before he could see solutions and provide a new reference point. However, the waiting time was acceptable for him.

The DM provided the first reference point as

$$\hat{z}_1 = (84, 90, 86)$$

and assumed he would see solutions that could reach it. It was selected based on the DM's expertise, as objective function values that were expected (based on prior knowledge of similar pumps and hydraulic design in general) to be achieved. As said, the method generated 15 new solutions and they were presented to the DM in the form of a table (see Table A2 in the appendix) and a parallel coordinate plot, see Figure 5(a).

However, the reference point $\hat{z}_1$, depicted in green (dotted), was not reached. Therefore, the DM decided to keep the reference point unchanged ($\hat{z}_2 = \hat{z}_1$) and continue seeking improved solutions in the same subregion of the objective space.

The next 15 solutions generated after the second interaction are depicted in Figure 5(b). As can be observed, they still did not reach the reference point $\hat{z}_2$. Therefore, the DM decided to continue still with the same reference point ($\hat{z}_3 = \hat{z}_2$). Finally, after the third interaction, two solutions dominating the reference point $\hat{z}_3$ were reached:

$$(84.7, \ 90.2, \ 87.6), \quad (85.0, \ 90.1, \ 87.9).$$

They are highlighted in dark blue (dashed) in Figure 5(c).

In the fourth interaction, the DM provided a new reference point with a higher value for the first objective:

$$\hat{z}_4 = (86, 90, 86).$$

This choice was based on the assumption that the efficiency at the sub-optimal flow rates could be improved without a big sacrifice in the other regions of the working range. Based on the new reference point, the next 15 solutions were generated and shown to the DM, see Figure 5(d).

In the fifth interaction, he decided to keep the reference point the same as in the fourth interaction ($\hat{z}_5 = \hat{z}_4$). The solutions generated after the fifth interaction are depicted in Figure 5(e). Two solutions (depicted in dark blue (dashed)) were very close to the reference point $\hat{z}_5$:

$$(86.1, \ 89.9, \ 87.4), \quad (86.0, \ 90.0, \ 87.7).$$

Based on the solutions obtained, the DM was interested in further improving the first objective value and finding a good compromise between the efficiencies in the lower and higher flow rates. Therefore,
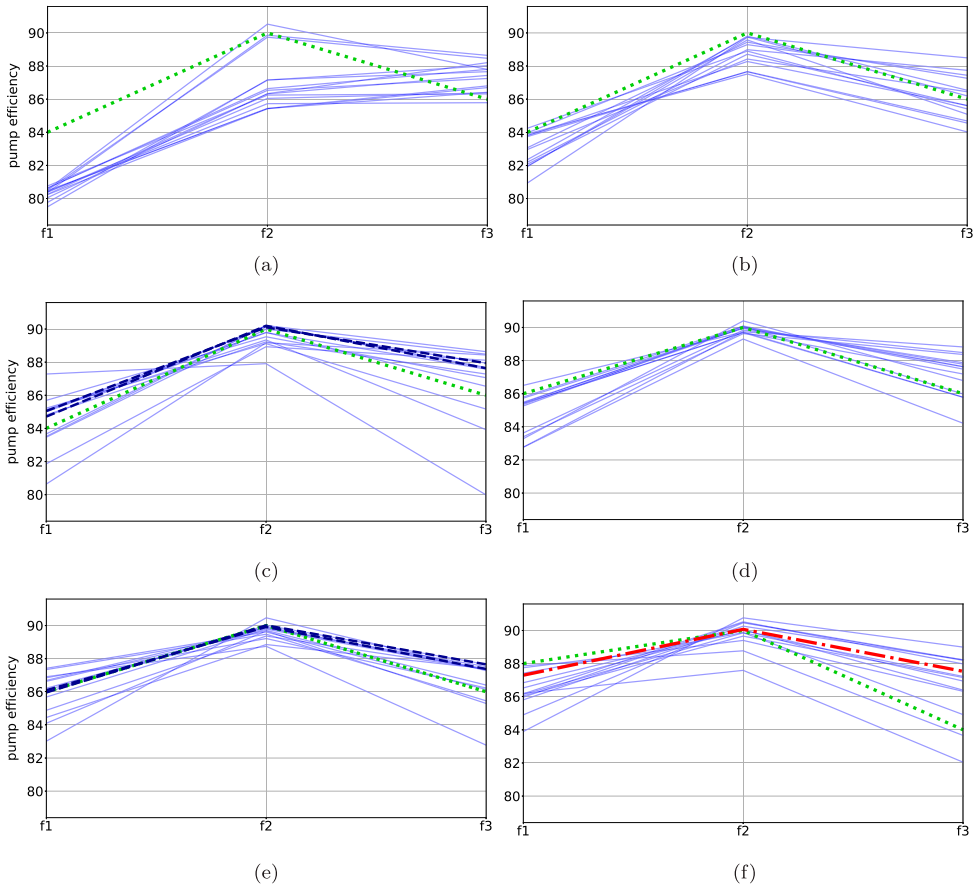
Figure 5. Solutions generated after the first (a), second (b), third (c), fourth (d), fifth (e) and sixth (f) interaction with the DM. Solutions are shown in blue (solid), reference points in green (dotted), solutions dominating the reference point (c) or close to the reference point (e) in dark blue (dashed) and the final solution in red (dash-dotted) (f).

in the sixth interaction, the DM set the reference point as

$$\hat{z}_6 = (88, 90, 84).$$

New solutions were generated as shown in Figure 5(f). The DM was satisfied with the solutions generated, and selected his final, most preferred, solution:

$$(87.3, \ 90.1, \ 87.5).$$

The final solution is highlighted in red (dash-dotted) in Figure 5(f). The corresponding (normalized values of the) design variables are

$$
\begin{aligned}
\mathbf{x} = \ & (0.0021, 0.3308, 0.4075, 0.4597, 0.1449, 0.4230, \\
& 0.8698, 0.9994, 0.2729, 0.1826, 0.7991, 0.8597, \\
& 0.0001, 0.1257, 0.2273, 0.7239, 0.5580, 0.8652, \\
& 0.6757, 0.3823, 0.6637, 0.2028).
\end{aligned}
$$

The 3D model of the final pump design is shown in Figure 6.

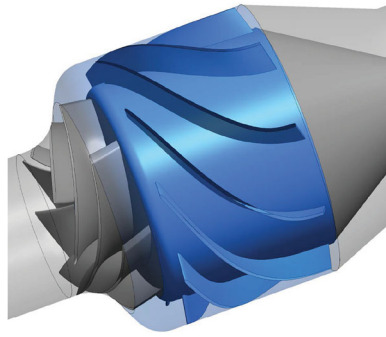**Figure 6.** The final solution, selected by the DM. The optimized stator is in blue. (Colour figure online).

The design variables of all solutions shown to the DM in the last interaction, and the design variables of the selected solution, are shown as a parallel coordinate plot in Figure A1(b) in the appendix.

The solution process was terminated in the seventh interaction by the DM, *i.e.* after 30 iterations. The number of new solutions that were generated and evaluated with the three computationally expensive objective functions was 90. The initial dataset consisted of 37 successfully generated solutions. Therefore, the total number of computationally expensive function evaluations was $37 + 90 = 127$. A positive outcome is that no failed samples were generated during the solution process. This may be explained by the model management, which selected solutions close to the reference point and thus successfully led solutions into the region of interest to the DM, and by the fact that the DM, as a domain expert, had good knowledge of the problem and its limitations, and thus provided realistic preferences that directed the search to successful regions. These two aspects of the model management followed the DM's preferences and, on the basis of the DM's expertise, increased the probability of avoiding infeasible regions and produced solutions with parameters leading to viable geometries.

5. Discussion

Originally, in Krátký (2020), the pump design problem was scalarized into a problem with a single objective function and solved by a surrogate-based stochastic radial basis function method (SRBF) (Regis and Shoemaker 2009) with an implementation in MATLAB.[1] SRBF needed 300 iterations (300 function evaluations) to solve the problem. As mentioned earlier, K-RVEA was able to find a comparable solution with only 45 iterations (135 function evaluations). This shows the advantage of using multiobjective optimization methods with proper model management, which can save weeks of computation time.

As demonstrated in Section 3, a-priori K-RVEA could not provide solutions that followed the DM's preferences. In the model management of this method, one of the criteria to update the models is the contribution of each solution to the diversity of the Pareto front approximation represented. This criterion may not be ideal from the perspective of the DM's preferences, because the method may ignore solutions that reflect the DM's preferences and choose those that are far but improve the diversity. Moreover, the model management prioritizes solutions with a high uncertainty. This means that the CFD simulator results and the predicted values of the surrogate models can differ significantly.

On the other hand, interactive K-RVEA generated promising solutions reflecting the DM's preferences. A-priori K-RVEA and interactive K-RVEA handle the DM's preferences in the same way during the optimization process. However, their main difference is in their model management, where interactive K-RVEA considers preferences. First, some solutions that reflect the DM's preferences the

best are selected, and those are chosen that have a low uncertainty. In this way, designs to be evaluated with CFD simulations have a high likelihood of reflecting preferences. This seems to be a reason that interactive K-RVEA generated solutions that followed the DM's preferences better than a-priori K-RVEA.

Moreover, even though some solutions of K-RVEA dominated those of interactive K-RVEA's with a small margin, the differences were insignificant (considering the accuracy of the CFD simulations). That is, the DM was equally satisfied with both solutions. In the literature, solution processes with evolutionary algorithms are often repeated many times to compensate for the uncertainty involved. Repetitions were not possible here owing to the high computation cost of the simulations and limited resources.

In many interactions, the DM decided to continue with the same reference point since he felt that better solutions could be found and he was right. This means that the method had not yet converged near the Pareto front and subsequent generations of the evolutionary algorithm could produce solutions that dominated previous ones. This means that more iterations should have been conducted between interactions. Thus, the stopping criteria should be re-considered.

When the pump design problem was solved with K-RVEA, 135 simulation calls were needed in addition to the initial population before termination (Krátký 2020). However, interactive K-RVEA reached the most preferred solution after 30 iterations, which means 90 simulation calls after the initial population was generated. Since the problem is extremely computationally expensive, this means saving more than 11 days in computation time. Additionally, interactive K-RVEA did not generate any failed samples while K-RVEA generated 36 failed samples (Krátký 2020). The ability to avoid any failed samples cannot be guaranteed in general. However, there is a causal relationship between model management led by a DM with advanced problem domain expertise and the number of failed samples generated, as shown on the pump design problem.

6. Conclusions

This article dealt with solving a computationally extremely expensive pump design problem. Previously, the problem had been solved with a surrogate-based multiobjective optimization method K-RVEA that did not incorporate the DM's preferences. In other words, it was an a-posteriori method. First, an a-priori version of that method was applied but it was not a success. On the other hand, very encouraging results were obtained with a surrogate-assisted interactive method called interactive K-RVEA. The DM found the solutions very satisfactory and computational resources were significantly saved when compared to an a-posteriori method.

This problem demonstrated that interactive methods are helpful in solving real-world problems. Incorporating the DM's preferences iteratively during the solution process had the following three main advantages as observed in the pump design problem.

(1) Computational resources were saved (compared to an a-posteriori method).
(2) Solutions generated reflected the DM's preferences well (compared to an a-priori method).
(3) Simulation failures were avoided.

Moreover, the role of model management in surrogate-assisted methods was presented and the need to incorporate the DM's preferences not only in the solution process but also in the model management was discussed.

One possible future research direction would be to design a new interactive method to fit the specific needs of this type of problem explicitly with novel model management. Moreover, more intelligent stopping criteria for interactions and incorporating different types of preference (not limiting the DM to providing only reference points) would be further research directions to explore.

Note

1. https://ccse.lbl.gov/people/julianem/

Data availability statement

The data that support the findings of this study are available upon reasonable request to the corresponding author. Interactive K-RVEA is implemented in the open-source software DESDEO (Misitano *et al.* 2021) https://github.com/industrial-optimization-group.

Disclosure statement

No potential conflict of interest was reported by the author(s).

ORCID

Jana Burkotová http://orcid.org/0000-0002-8571-2597
Pouya Aghaei Pour http://orcid.org/0000-0002-2689-2567
Tomáš Krátký http://orcid.org/0000-0003-1514-4202
Kaisa Miettinen http://orcid.org/0000-0003-1013-4689

References

Aghaei Pour, P., T. Rodemann, J. Hakanen, and K. Miettinen. 2022. "Surrogate Assisted Interactive Multiobjective Optimization in Energy System Design of Buildings." *Optimization and Engineering* 23 (1): 303–327. https://doi.org/10.1007/s11081-020-09587-8.

Alizadeh, R., J. K. Allen, and F. Mistree. 2020. "Managing Computational Complexity Using Surrogate Models: A Critical Review." *Research in Engineering Design* 31 (3): 275–298. https://doi.org/10.1007/s00163-020-00336-7.

Cai, X., L. Gao, and X. Li. 2020. "Efficient Generalized Surrogate-Assisted Evolutionary Algorithm for High-Dimensional Expensive Problems." *IEEE Transactions on Evolutionary Computation* 24 (2): 365–379. https://doi.org/10.1109/TEVC.2019.2919762.

Cheng, R., Y. Jin, M. Olhofer, and B. Sendhoff. 2016. "A Reference Vector Guided Evolutionary Algorithm for Many-Objective Optimization." *IEEE Transactions on Evolutionary Computation* 20 (5): 773–791. https://doi.org/10.1109/TEVC.2016.2519378.

Chugh, T., Y. Jin, K. Miettinen, J. Hakanen, and K. Sindhya. 2018. "A Surrogate-Assisted Reference Vector Guided Evolutionary Algorithm for Computationally Expensive Many-Objective Optimization." *IEEE Transactions on Evolutionary Computation* 22 (1): 129–142. https://doi.org/10.1109/TEVC.2016.2622301.

Chugh, T., T. Kratky, K. Miettinen, Y. Jin, and P. Makkonen. 2019. "Multiobjective Shape Design in a Ventilation System with a Preference-Driven Surrogate-Assisted Evolutionary Algorithm." In *Proceedings of the Thirteenth International Conference on Genetic and Evolutionary Computing (GECCO 2019)*, 1147–1155. New York: Association for Computing Machinery (ACM). https://doi.org/10.1145/3321707.3321745.

Chugh, T., K. Sindhya, J. Hakanen, and K. Miettinen. 2019. "A Survey on Handling Computationally Expensive Multiobjective Optimization Problems with Evolutionary Algorithms." *Soft Computing* 23 (9): 3137–3166. https://doi.org/10.1007/s00500-017-2965-0.

Deb, K. 2001. *Multi-Objective Optimization Using Evolutionary Algorithms*. Chichester, UK: Wiley.

Fracassi, A., R. De Donno, A. Ghidoni, and P. M. Congedo. 2022. "Shape Optimization and Uncertainty Assessment of a Centrifugal Pump." *Engineering Optimization* 54 (2): 200–217. https://dx.doi.org/10.1080/0305215X.2020.1858075.

Gan, X., J. Pei, W. Wang, S. Yuan, and B. Lin. 2023. "Application of a Modified MOPSO Algorithm and Multi-Layer Artificial Neural Network in Centrifugal Pump Optimization." *Engineering Optimization* 55 (4): 580–598. https://dx.doi.org/10.1080/0305215X.2021.2015585.

Hakanen, J., T. Chugh, K. Sindhya, Y. Jin, and K. Miettinen. 2016. "Connections of Reference Vectors and Different Types of Preference Information in Interactive Multiobjective Evolutionary Algorithms." In *Proceedings of the 2016 IEEE Symposium Series on Computational Intelligence (SSCI)*, 1–8. Athens, Greece. https://dx.doi.org/10.1109/SSCI.2016.7850220.

Jaiswal, A. K., M. H. Siddique, A. R. Paul, and A. Samad. 2022. "Surrogate-Based Design Optimization of a Centrifugal Pump Impeller." *Engineering Optimization* 54 (8): 1395–1412. https://dx.doi.org/10.1080/0305215X.2021.1932867.

Jin, Y. 2011. "Surrogate-Assisted Evolutionary Computation: Recent Advances and Future Challenges." *Swarm and Evolutionary Computation* 1 (2): 61–70. https://doi.org/10.1016/j.swevo.2011.05.001.

Jin, Y., H. Wang, and C. Sun. 2021. *Data-Driven Evolutionary Optimization — Integrating Evolutionary Computation, Machine Learning and Data Science*. Cham, Switzerland: Springer. https://doi.org/10.1007/978-3-030-74640-7.

Krátký, T. 2020. "Shape Optimization of Hydraulic Surfaces of the Impeller and Stator Parts of Hydrodynamic Pumps." PhD diss., Faculty of Science, Palacký University Olomouc.

Lim, D., Y. Jin, Y. S. Ong, and B. Sendhoff. 2010. "Generalizing Surrogate-Assisted Evolutionary Computation." *IEEE Transactions on Evolutionary Computation* 14 (3): 329–355. https://doi.org/10.1109/TEVC.2009.2027359.

Miettinen, K. 1999. *Nonlinear Multiobjective Optimization*. Boston, MA: Kluwer.

Miettinen, K., J. Hakanen, and D. Podkopaev. 2016. "Interactive Nonlinear Multiobjective Optimization Methods." In *Multiple Criteria Decision Analysis*, edited by S. Greco, M. Ehrgott, and J. Figueira, 2nd ed., 927–976. New York: Springer. https://doi.org/10.1007/978-1-4939-3094-4_22.

Miettinen, K., F. Ruiz, and A. P. Wierzbicki. 2008. "Introduction to Multiobjective Optimization: Interactive Approaches." In *Multiobjective Optimization*, edited by J. Branke, K. Deb, K. Miettinen, and R. Slowinski, 27–57. Berlin: Springer. https://doi.org/10.1007/978-3-540-88908-3_2.

Misitano, G., B. S. Saini, B. Afsar, B. Shavazipour, and K. Miettinen. 2021. "DESDEO: The Modular and Open Source Framework for Interactive Multiobjective Optimization." *IEEE Access* 9:148277–148295. https://doi.org/10.1109/ACCESS.2021.3123825.

Regis, R., and C. Shoemaker. 2009. "Parallel Stochastic Global Optimization Using Radial Basis Functions." *INFORMS Journal on Computing* 21 (3): 411–426. https://doi.org/10.1287/ijoc.1090.0325.

Sacks, J., S. B. Schiller, and W. J. Welch. 1989. "Designs for Computer Experiments." *Technometrics* 31 (1): 41–47. https://doi.org/10.1080/00401706.1989.10488474.

Wang, H., L. Feng, Y. Jin, and J. Doherty. 2021. "Surrogate-Assisted Evolutionary Multitasking for Expensive Minimax Optimization in Multiple Scenarios." *IEEE Computational Intelligence Magazine* 16 (1): 34–48. https://doi.org/10.1109/MCI.2020.3039067.

Wierzbicki, A. P. 1982. "A Mathematical Basis for Satisficing Decision Making." *Mathematical Modelling* 3 (5): 391–405. https://doi.org/10.1016/0270-0255(82)90038-0.

Zhao, F., Z. Zeng, H. Q. Liu, and J. L. Fan. 2019. "A Kriging-Assisted Reference Vector Guided Multi-Objective Evolutionary Fuzzy Clustering Algorithm for Image Segmentation." *IEEE Access* 7:21465–21481. https://doi.org/10.1109/ACCESS.2019.2897597.

Zitzler, E., and L. Thiele. 1998. "Multiobjective Optimization Using Evolutionary Algorithms — A Comparative Case Study." In *Proceedings of the 5th International Conference on Parallel Problem Solving from Nature (PPSN V)*, 292–304. Berlin: Springer. https://dl.acm.org/doi/10.5555/645824.668610.

Appendix

This appendix is devoted to additional information. A description of the design variables of the pump design problem is given and parallel coordinate plots of solutions generated using K-RVEA and interactive K-RVEA in the design space are provided. Moreover, details of the solution process with interactive K-RVEA are provided in the form of a table.

Design variables

The pump design problem is described with 22 design variables that correspond to the geometry parameters. Their description and bounds are presented in Table A1.

For easier comparison, normalized design variable values in parallel coordinate plots are presented. In Figure A1(a), all 17 non-dominated solutions generated by K-RVEA are depicted in the design space in blue (solid) and the final solution in red (dash-dotted).

Figure A1(b) shows design variables of solutions generated by the interactive K-RVEA method. The last 15 solutions that were shown to the DM before he decided to terminate the solution process are depicted in blue (solid) and the final solution in red (dash-dotted). As can be seen, K-RVEA covered a wider range in the design space, as expected, whereas interactive K-RVEA concentrated the search in a specific region that reflected the DM's reference point.

Table A1. Description of the design variables of the pump design problem.

Index	Range	Description
1	[20, 30]	Beta angle - hub - leading edge (degrees)
2	[0.22, 0.72]	Beta angle - hub - relative value at 25%
3	[0.22, 0.76]	Beta angle - hub - relative value at 50%
4	[0.25, 0.78]	Beta angle - hub - relative value at 75%
5	[−5, 0]	Beta angle - hub - value at 95% - difference from trailing edge value (degrees)
6	[85, 90]	Beta angle - hub - trailing edge (degrees)
7	[355, 380]	Outlet diameter - hub (mm)
8	[450, 600]	Meridional length - hub (mm)
9	[15, 45]	Leading edge position - hub - distance from the inlet (mm)
10	[15, 50]	Trailing edge position - hub - distance to the outlet (mm)
11	[−10, 10]	Sweep angle (defined at the shroud) (degrees)
12	[16, 26]	Beta angle - shroud - leading edge (degrees)
13	[0.25, 0.76]	Beta angle - shroud - relative value at 25%
14	[0.22, 0.7]	Beta angle - shroud - relative value at 50%
15	[0.25, 0.76]	Beta angle - shroud - relative value at 75%
16	[−5, 0]	Beta angle - shroud - value at 95% - difference from trailing edge value (degrees)
17	[85, 90]	Beta angle - shroud - trailing edge (degrees)
18	[450, 600]	Meridional length - shroud (mm)
19	[15, 60]	Leading edge position - shroud - distance from the inlet (mm)
20	[15, 50]	Trailing edge position - shroud - distance to the outlet (mm)
21	[27, 35]	Outflow angle - hub (degrees)
22	[−15, 5]	Outflow angle - shroud - relative to the hub angle (degrees)

Solution process with interactive K-RVEA

The solution process with interactive K-RVEA was described in Section 4. In each interaction, the DM had 15 new solutions at his disposal that were generated in the last 5 iterations (another DM might have preferred to see fewer solutions at a time). The solutions were presented in the form of a parallel coordinate plot and a table. Table A2 summarizes the solution process. The DM selected the final solution in the seventh interaction, *i.e.* after 30 iterations. The final solution is marked in bold type with an asterisk, other solutions important to the DM mentioned in Section 4 are marked in bold type without an asterisk, *i.e.* solutions that outperformed $\hat{z}_3$ after the third interaction, and solutions that were very close to $\hat{z}_5$ after the fifth interaction.

Table A2. Summary of the solution process with interactive K-RVEA.

| Iteration	f_1	f_2	f_3	Iteration	f_1	f_2	f_3	Iteration	f_1	f_2	f_3
1	80.3	86.3	87.4	11	85.7	89.5	87.1	21	84.9	89.4	87.4
	80.6	86.6	87.8		84.8	89.2	87.3		86.8	89.2	86.2
	80.6	86.5	87.2		80.6	89.2	88.4		86.2	89.8	87.5
2	80.4	86.0	86.4	12	83.7	90.2	87.7	22	87.3	89.5	85.5
	80.4	85.7	85.8		85.0	89.8	86.5		83.0	90.5	87.3
	80.8	86.2	86.3		85.1	89.3	83.9		84.1	89.7	85.3
3	80.5	85.4	86.8	13	81.9	88.9	88.1	23	**86.1**	**89.9**	**87.4**
	80.2	85.4	86.7		**84.7**	**90.2**	**87.6**		86.9	88.7	82.8
	80.5	85.4	86.4		83.5	90.2	88.6		86.6	89.7	86.4
4	80.1	86.3	88.2	14	85.2	89.1	85.2	24	84.4	88.9	87.4
	79.5	87.2	87.7		85.1	89.8	87.6		87.4	89.6	86.1
	79.8	87.1	88.0		87.3	87.9	80.0		86.7	89.9	87.3
5	80.5	89.7	88.5	15	83.5	90.0	88.5	25	**86.0**	**90.0**	**87.7**
	80.4	89.9	88.6		**85.0**	**90.1**	**87.9**		85.7	89.8	87.3
	80.6	90.5	87.8		85.2	89.7	85.1		82.0	88.6	85.5
6	83.8	87.7	84.6	16	83.3	90.4	86.8	26	87.8	89.4	86.3
	83.7	87.7	84.7		85.4	89.7	87.2		83.9	90.8	89.0
	83.9	87.5	84.0		85.3	90.1	87.5		86.0	90.5	88.2
7	82.0	89.6	85.1	17	85.3	90.0	87.8	27	86.2	87.6	82.0
	82.3	89.8	86.2		83.6	89.8	88.5		86.5	90.0	86.4
	82.1	88.7	85.6		85.4	89.7	85.8		85.8	90.3	88.2
8	81.9	89.8	86.5	18	85.5	90.0	87.6	28	**87.3***	**90.1***	**87.5***
	83.8	88.3	85.6		83.4	89.6	88.4		86.1	89.7	87.2
	83.0	88.4	86.4		85.8	90.1	86.0		86.1	89.9	87.2
9	83.1	89.4	87.3	19	82.8	89.3	84.2	29	86.2	90.5	88.0
	82.2	89.0	87.7		86.5	89.7	85.8		87.4	89.9	84.9
	84.2	88.9	85.4		85.5	90.0	87.6		84.9	90.5	88.2
10	80.9	89.7	88.5	20	85.7	89.9	87.9	30	87.9	88.8	83.7
	84.0	89.3	87.4		82.8	89.7	88.8		86.8	90.0	86.9
	84.4	89.0	84.5		86.5	89.4	86.6		86.3	90.4	88.1
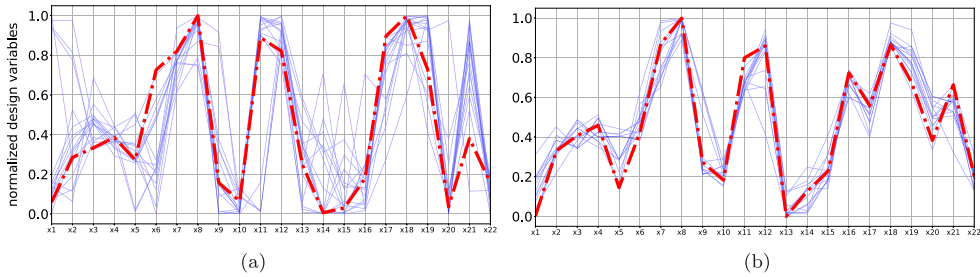
Figure A1. (a) Non-dominated solutions obtained with K-RVEA in the design space are in blue (solid). The final design is shown in red (dash-dotted). (b) Solutions obtained with interactive K-RVEA after the sixth interaction in the design space are in blue (solid). The final design is shown in red (dash-dotted).