Author(s): Zhou-Kangas, Yue; Miettinen, Kaisa; Sindhya, Karthik

Title: Interactive Multiobjective Robust Optimization with NIMBUS

Year: 2018

Version: Accepted version (Final draft)

Interactive Multiobjective Robust Optimization with NIMBUS

Yue Zhou-Kangas, Kaisa Miettinen, and Karthik Sindhya

University of Jyvaskyla, Faculty of Information Technology,
P.O.Box 35 (Agora), FI-40014, University of Jyvaskyla, Finland
yue.y.zhou-kangas@jyu.fi

Abstract. In this paper, we introduce the MuRO-NIMBUS method for solving multiobjective optimization problems with uncertain parameters. The concept of set-based minmax robust Pareto optimality is utilized to tackle the uncertainty in the problems. We separate the solution process into two stages: the pre-decision making stage and the decision making stage. We consider the decision maker's preferences in the nominal case, i.e., with the most typical or undisturbed values of the uncertain parameters. At the same time, the decision maker is informed about the objective function values in the worst case to support her/him to make an informed decision. To help the decision maker to understand the behaviors of the solutions, we visually present the objective function values. As a result, the decision maker can find a preferred balance between robustness and objective function values under the nominal case.

Keywords: Multiple criteria decision making, uncertainty, robustness, interactive methods, robust Pareto optimality

1 Introduction

Many real-life optimization problems involve multiple (conflicting) objectives. Multiobjective optimization methods (see e.g., [11] and [18]) solve these problems by optimizing the conflicting objectives simultaneously. For multiobjective optimization problems, there usually is a set of mathematically equally good solutions with different trade-offs among the multiple objectives. These solutions are called Pareto optimal solutions. In most cases, only one Pareto optimal solution is chosen as the final solution to implement. This solution is usually found by utilizing preferences of a decision maker, who is an expert in the problem domain.

Different types of methods can be identified depending on the role of the decision maker [11]. In interactive multiobjective optimization methods [3] , the decision maker actively directs the solution process towards a most preferred solution by iteratively specifying her/his preferences. With an active involvement, which is not possible in other types of methods, the decision maker can gradually learn about the problem and its feasible solutions as well as how attainable her/his preferred solutions are. In this way, interactive methods can best support the decision maker to find the most preferred solution.

In addition to multiple objectives, the presence of uncertainty in real-life optimization problems should be considered due to imprecise data, uncertain operation environments, and uncertain future developments, etc. The uncertainty can be reflected in parameters or decision variables in problem formulations. In this paper, we concentrate on problems with uncertain parameters in objective functions. With different realizations of uncertain parameters, the corresponding outcomes (i.e., objective function values) are different.

On one hand, without considering the uncertainty, the outcome corresponding to a deterministic Pareto optimal solution can become very bad when the uncertain parameters realize differently. Many robustness concepts have been defined for multiobjective optimization problems (see e,g., [9] [19]). They guarantee the immunity of solutions to uncertainty by transforming uncertain problems to deterministic ones with respect to the worst case. On the other hand, the outcomes in the nominal case are very important for the decision maker, because the nominal case describes the most typical behavior of uncertain parameters. In addition, the robustness and quality of solutions, i.e., the outcome in the nominal case, usually conflict with each other [1]. In other words, objective function values of a robust Pareto optimal solution are usually not as good as those of a deterministic Pareto optimal solution in the nominal case.

When considering uncertainty, the decision maker faces the challenge of making a decision with respect to different possible outcomes because of different realizations of uncertain parameters. Considering multiple possible realizations simultaneously can be too challenging for the decision maker. In addition, it is desirable for the decision maker to find a preferred balance between robustness and quality of the solutions. With the help of multiobjective robust optimization, we can guarantee the robustness of solutions by finding the best solutions with respect to the worst case but at the same time, the decision maker needs support to find a most preferred balance between robustness and quality of solutions.

In the literature, most research efforts have been devoted to different definitions of robust Pareto optimality and only a few solution methods have been developed (e.g., in [5] and [10]). In addition, in [2], necessary and sufficient conditions for scalarizing functions with some special properties are discussed, which can be used to transform a multiobjective optimization problem to a single-objective one. In [7], [8], [14], and [15], interactive methods have been utilized to find a final solution for multiobjective optimization problems with uncertainty.

In [7] and [8], a robust version of the augmented weighted Chebyshev method [17] was developed for multiobjective linear optimization problems by extending the concept of the budget of uncertainty [1] to multiobjective optimization problems. Uncertainty was tackled in a so-called all-in-one approach in [14], where the decision maker considers all possible realizations of uncertain parameters simultaneously. During the solution process, the decision maker chooses the possible realizations to concentrate on and formulates her/his preferences with respect to them. In [15], the decision maker is expected to specify weights to alter the relative importance of objectives and robustness when they are combined to formulate a single-objective optimization problem.

In this paper, we develop an interactive method called MuRO-NIMBUS to better support the decision maker. The MuRO-NIMBUS method integrates the concept of set-based minmax robustness [5] into the NIMBUS framework, which to the best of our knowledge, is the first interactive method for supporting a decision making to find set-based minmax robust Pareto optimal solutions.

The properties of desirable interactive methods were summarized in [16] in terms of understandability, easiness to use, and features of being supportive. In order to ensure those properties in MuRO-NIMBUS, we first guarantee the robustness of solutions by utilizing the set-based minmax robust Pareto optimality to find a set of best possible solutions in the worst case. For this step, we develop a robust achievement scalarizing function approach, which can also be used independently. Then we incorporate the preferences of the decision maker to find a solution corresponding to a most preferred outcome in the nominal case. At the same time, the decision maker is informed about the worst possible values. In order to support the decision maker to understand the solution in terms of its objective function values in the nominal case and the objective function values in the worst case, we augment the value path visualization (see e.g., [6]) to visually present different types of information. In this way, we can support the decision maker to grasp a total balance in the robustness and quality of solutions during the solution process.

By applying MuRO-NIMBUS, the decision maker is not expected to consider all possible realizations of the uncertain parameters simultaneously as in [14]. Unlike in [7] and [8] where solutions once discarded cannot be recovered, the decision maker can move freely from one robust Pareto optimal solution to another. Instead of providing preferences as weights which do not have concrete meanings as in [15], MuRO-NIMBUS allows the decision maker to concretely consider the objective function values of a more desired solution.

The rest of the paper is organized as follows: in the next section, we introduce some basic concepts. In Section 3, we introduce MuRO-NIMBUS. We simulate the solution process of a multiobjective ship design problem as a numerical example in Section 4 to demonstrate the application of the new method. Finally, we conclude the paper in Section 5.

2 Basic Concepts

2.1 Deterministic Multiobjective Optimization

A deterministic multiobjective optimization problem is of the form

$$\begin{array}{ll} \text{minimize or maximize} & \{f_1(\boldsymbol{x}), ..., f_k(\boldsymbol{x})\} \\ \text{subject to} & \boldsymbol{x} \in \mathcal{X}, \end{array} \tag{1}$$

involving objective functions (objectives) $f_i : \mathcal{X} \to \mathbb{R}$ to be simultaneously optimized, where $1 \leq i \leq k$ and $k \geq 2$. Objective vectors $\boldsymbol{f}(\boldsymbol{x}) = (f_1(\boldsymbol{x}), ..., f_k(\boldsymbol{x}))^T$

consist of objective function values which are the images of decision vectors $\boldsymbol{x} = (x_1, x_2, ..., x_n)^T$. Decision vectors belong to the nonempty feasible set $\mathcal{X} \subset \mathbb{R}^n$ and their components are called decision variables. In this paper, we refer to decision vectors as solutions and objective vectors as outcomes or objective function values of solutions. For two feasible solutions, we say a solution dominates the other when the value of at least one of the objectives is better and others are at least as good as that of the other. For simplicity, we assume that the objective functions are to be minimized.

Definition 1. *A solution $\boldsymbol{x}^* \in \mathcal{X}$ is said to be Pareto optimal or efficient if there does not exist another solution $\boldsymbol{x} \in \mathcal{X}$ such that $f_i(\boldsymbol{x}) \leq f_i(\boldsymbol{x}^*)$ for all $i = 1, ..., k$ and $f_j(\boldsymbol{x}) < f_j(\boldsymbol{x}^*)$ for at least one j.*

With the help of the nonnegative ordering cone $\mathbb{R}^k_{\geqq} = \{\boldsymbol{z} \in \mathbb{R}^k | z_i \geq 0 \text{ for } i = 1, ..., k\}$, we say that $\boldsymbol{x}^*$ is Pareto optimal if there does not exist $\boldsymbol{x} \in \mathcal{X}$ such that $\boldsymbol{f}(\boldsymbol{x}) \in \boldsymbol{f}(\boldsymbol{x}^*) - \mathbb{R}^k_{\geqq}$. We refer to the set of Pareto optimal solutions as the Pareto optimal set.

For (1), the set of Pareto optimal solutions usually contains more than one element. For the decision maker, it is often useful to know the ranges of the objective function values in the Pareto optimal set. The ranges are given by the ideal objective vector $\boldsymbol{z}^* = (z_1^*, ..., z_k^*)^T$ and the nadir objective vector $\boldsymbol{z}^{nad} = (z_1^{nad}, ..., z_k^{nad})^T$. The ideal objective vector is formed by individual optima of each objective function in the feasible set. For computational reasons, we use the utopian objective vector $\boldsymbol{z}^{**}$, which is strictly better than $\boldsymbol{z}^*$. In practice, z_i^{**} is set as $z_i^* - a$ for $i = 1, ..., k$, where $a > 0$ is a small scalar. The nadir objective vector, which represents the worst objective function values, can be approximated for example by a so-called pay-off table (see [11] for further details). If the objective function values have different magnitudes, $\boldsymbol{z}^{nad}$ and $\boldsymbol{z}^{**}$ can be used to normalize them for computing purposes.

For calculating Pareto optimal solutions, one approach is to scalarize, i.e., to formulate a single objective optimization problem such that its optimal solution is a Pareto optimal solution for (1). In this, a single objective solver which is appropriate for the characteristics of the problem must be used. The achievement scalarizing function [20] is one of the widely used scalarizing functions. In this paper, we consider the achievement scalarizing function of the following form:

$$\begin{aligned} &\text{minimize} \quad \max{}_i \;\; [w_i(f_i(\boldsymbol{x}) - \bar{z}_i)] + \rho \sum_{i=1}^{k} w_i(f_i(\boldsymbol{x}) - \bar{z}_i) \\ &\text{subject to} \quad \boldsymbol{x} \in \mathcal{X}, \end{aligned} \tag{2}$$

where ρ is a small scalar binding the trade-offs, $\bar{\boldsymbol{z}}$ is a reference point and its component $\bar{z}_i$ is the aspiration level which represents the desired value of the objective function f_i given by the decision maker. The positive weight vector $\boldsymbol{w}$ sets a direction toward which the reference point is projected onto the Pareto optimal set.

As discussed in the literature (e.g., [3], [11], and [20]), the optimal solution of (2) is a Pareto optimal solution for (1) and any Pareto optimal solution with trade-offs bounded by ρ can be found by changing $\bar{\boldsymbol{z}}$. The achievement scalarizing function has many advantages, for example, the reference point can be feasible or infeasible and the problem can be convex or nonconvex.

2.2 Uncertain Multiobjective Optimization Problems and Set-based Minmax Robustness

For multiobjective optimization problems with uncertain parameters, given an uncertainty set $\mathcal{U} \subseteq \mathbb{R}^m$, the uncertain multiobjective optimization problem is given as a collection of deterministic multiobjective optimization problems:

$$\left\{ \begin{array}{ll} \text{minimize} & \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi}) \\ \text{subject to} & \boldsymbol{x} \in \mathcal{X} \end{array} \right\}_{\boldsymbol{\xi} \in \mathcal{U}.} \tag{3}$$

Every problem in the collection is called an instance, which is characterized by a particular element $\boldsymbol{\xi} \in \mathcal{U}$. Depending on different realized values of $\boldsymbol{\xi}$, a decision vector can have different corresponding outcomes. As a result, we have a set of outcomes corresponding to a feasible decision vector. We denote the set of outcomes (i.e., the objective vectors) of a solution $\boldsymbol{x} \in \mathcal{X}$ for all $\boldsymbol{\xi} \in \mathcal{U}$ as $f_{\mathcal{U}}(\boldsymbol{x}) = \{f_{\mathcal{U}}(\boldsymbol{x}, \boldsymbol{\xi}) : \boldsymbol{\xi} \in \mathcal{U}\}$ as in [5].

As briefly mentioned, among all the possible realizations of uncertain parameters, the nominal case $\hat{\boldsymbol{\xi}}$ describes the most typical behavior of the uncertain parameters. It usually comes from previous experiences or the expert knowledge of the decision maker. The worst case describes the situation where the objective functions attain their worst values within $\mathcal{U}$. For a fixed solution $\boldsymbol{x} \in \mathcal{X}$, we need to solve the following problem to find the worst case:

$$\begin{array}{ll} \text{maximize} & \{f_1(\boldsymbol{x}, \boldsymbol{\xi}), ..., f_k(\boldsymbol{x}, \boldsymbol{\xi})\} \\ \text{subject to} & \boldsymbol{\xi} \in \mathcal{U}. \end{array} \tag{4}$$

If the components of $\boldsymbol{\xi}$ do not relate to each other, there is a single worst case. If they are related to each other, there can be multiple worst cases. With the found worst case, the corresponding outcomes for the solution in question can be calculated. The worst case does not necessarily realize in practice, but the information on the outcomes provides the upper bounds of the objective function values of a solution within $\mathcal{U}$.

Analogously to the definition of Pareto optimality for deterministic problems, set-based minmax Pareto optimality was defined in [5] by comparing the sets of all outcomes corresponding to solutions.

Definition 2. *A solution $\boldsymbol{x}^*$ is a set-based minmax robust Pareto optimal solution for* (3)*, if there does not exist another $\boldsymbol{x} \in \mathcal{X}$ such that $f_{\mathcal{U}}(\boldsymbol{x}) \subseteq f_{\mathcal{U}}(\boldsymbol{x}^*) - \mathbb{R}^k_{\geq}$.*

In other words, a feasible solution $\boldsymbol{x}^*$ is a set-based minmax robust Pareto solution if there does not exist another feasible solution x such that for all outcomes $f(\boldsymbol{x}, \xi) \in f_{\mathcal{U}}(\boldsymbol{x})$, there exists an outcome $f(\boldsymbol{x^*}, \xi) \in f_{\mathcal{U}}(\boldsymbol{x}^*)$ with $f_i(\boldsymbol{x}, \xi) \leq f_i(\boldsymbol{x^*}, \xi)$ for all $i = 1, \cdots, k$. We apply this concept in MuRO-NIMBUS to be introduced. With this concept, the decision maker can understand that for all set-based minmax robust Pareto optimal solutions, there does not exist a feasible solution with better objective function values in every possible realization of the uncertain parameters.

By interpreting the supremum of a set as the set itself, the robust counterpart of (3) which transforms (3) to a deterministic problem to identify robust Pareto optimal solutions is given in [5] as:

$$\begin{aligned} \text{minimize} \quad & \sup_{\boldsymbol{\xi} \in \mathcal{U}} \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi}) \\ \text{subject to} \quad & \boldsymbol{x} \in \mathcal{X}. \end{aligned} \tag{5}$$

Set-based minmax robust Pareto optimal solutions are the best possible solutions in the worst case because they are obtained by minimizing the suprema of the sets of outcomes. As explained earlier, finding the worst case outcomes for a fixed solution $\boldsymbol{x} \in \mathcal{X}$ requires solving a multiobjective optimization problem with objectives to be maximized as (4). The notation sup in (5) denote the supreme of the outcome sets which is used to identify the worst case outcomes. For simplicity, in what follows, we refer to set-based minmax robust Pareto optimal solutions as robust Pareto optimal solutions.

2.3 Interactive Multiobjective Optimization

As mentioned, in interactive methods, the decision maker directs the solution process towards a most preferred solution by iteratively specifying her/his preferences. A typical solution process (e.g., [3]) starts by presenting a Pareto optimal solution to the decision maker. If the decision maker is satisfied, the final solution is found. If the decision maker is not satisfied, (s)he is expected to specify preferences for a more desired solution. Based on the preferences, a new Pareto optimal solution which satisfies the preferences best is found and presented to her/him. The solution process continues until the decision maker finds a most preferred solution.

NIMBUS ([11] and [13]) is a family of classification-based interactive methods. In NIMBUS, the decision maker can classify the objectives to indicate what kind of objective vector would be more preferred than the current one. The objective functions can be assigned to up to five different classes including:

$I^{<}$ for those to be improved (i.e., decreased in case of minimizing, increased in case of maximizing),
$I^{\leq}$ for those to be improved until some desired aspiration level $\hat{z}_i$,
$I^{=}$ for those that are satisfactory at their current level,
$I^{\geq}$ for those that may be impaired till a bound ϵ_i, and
$I^{\diamond}$ for those that are temporarily allowed to change freely.

If aspiration levels or bounds are used, the decision maker is expected to provide them. If the classification is feasible, i.e., the decision maker allows at least one of the objectives to be impaired to improve some objectives, a scalarizing problem is solved to find a new Pareto optimal solution reflecting the preferences. In the so-called synchronous NIMBUS method, up to four different solutions can be found in each iteration by solving different scalarizing problems. Since we have to consider robustness and quality of the solutions, we limit the cognitive load to the consideration of only one solution at a time. We will return later to the variant of the NIMBUS scalarizing problems we use in MuRO-NIMBUS.

MuRO-NIMBUS inherits the advantage of classifying the objectives. First, classification can remind the decision maker that it is not possible to improve all objective function values at the same time but impairment in some objective(s) must be allowed. Second, the decision maker deals with objective function values and (s)he does not need to connect different types of information. Instead, (s)he only needs to know what kind of changes (s)he desires for a new solution.

3 MuRO-NIMBUS

In this section, we introduce MuRO-NIMBUS. To be able to present it, we first introduce some building blocks that we need for designing the method.

3.1 Building Blocks of MuRO-NIMBUS

As a building block of MuRO-NIMBUS, we first present the robust version of (2). Based on it, we introduce the robust achievement scalarizing function (ASF) approach to calculate a set of robust Pareto optimal solutions.

Based on the concept of robust Pareto optimality and the robust counterpart as introduced in Section 2, the robust version of (2) can be formulated as:

$$\begin{array}{ll} \text{minimize} & \sup_{\boldsymbol{\xi}\in\mathcal{U}} \max_i [w_i(f_i(\boldsymbol{x},\boldsymbol{\xi}) - \bar{z}_i)] + \rho \sum_{i=1}^{k} w_i(f_i(\boldsymbol{x},\boldsymbol{\xi}) - \bar{z}_i) \\ \text{subject to} & \boldsymbol{x} \in \mathcal{X} \quad \text{for all} \quad \xi \in \mathcal{U}. \end{array} \tag{6}$$

Just like (2), the robust version involves a reference point and a weight vector. We now prove the sufficient condition of the robust Pareto optimality:

Theorem 1. *Given an uncertain multiobjective optimization problem* (3)*, if* $\boldsymbol{x}^*$ *is an optimal solution to* (6) *for some* $\bar{\boldsymbol{z}}$ *and* $\boldsymbol{w}$*, and* $\max_{\boldsymbol{\xi}\in\mathcal{U}} f_i(\boldsymbol{x},\boldsymbol{\xi})$ *exists for all* $\boldsymbol{x} \in \mathcal{X}$ *and for all* $i = 1, ..., k$*, then* $\boldsymbol{x}^*$ *is a robust Pareto optimal solution for* (3)*.*

Proof. Assume that $\boldsymbol{x^*}$ is not a robust Pareto optimal solution for (3). Then there exists $\boldsymbol{x'} \in \mathcal{X}$ such that $f_{\mathcal{U}}(\boldsymbol{x'}) \subseteq f(\boldsymbol{x^*}) - \mathbb{R}^k_{\geq}$. Based on Lemma 3.4 in [5], for all $\boldsymbol{\xi} \in \mathcal{U}$, there exists $\boldsymbol{\eta} \in \mathcal{U}$ such that $f_i(\boldsymbol{x'},\boldsymbol{\xi}) \leq f_i(\boldsymbol{x^*},\boldsymbol{\eta})$ for $i = 1, ..., k$ and for at least one i the strict inequality holds. Since $w_i > 0$, we have $\max_i [w_i(f_i(\boldsymbol{x'},\boldsymbol{\xi}) - \bar{z}_i)] + \rho \sum_{i=1}^{k} (f_i(\boldsymbol{x'},\boldsymbol{\xi}) - \bar{z}_i) < \max_i [w_i(f_i(\boldsymbol{x^*},\boldsymbol{\eta}) - \bar{z}_i)] + \rho \sum_{i=1}^{k} (f_i(\boldsymbol{x^*},\boldsymbol{\eta}) - \bar{z}_i)$, where for all $\boldsymbol{\xi} \in \mathcal{U}$ there exists a $\eta \in \mathcal{U}$ which satisfy the inequality. Further, we know that $\max_{\boldsymbol{\xi}\in\mathcal{U}} \max_i [w_i(f_i(\boldsymbol{x'},\boldsymbol{\xi}) - \bar{z}_i)] + \rho \sum_{i=1}^{k} (f_i(\boldsymbol{x'},\boldsymbol{\xi}) - \bar{z}_i) <$

$\max_{\boldsymbol{\eta}'\in\mathcal{U}} \max_i [w_i(f_i(\boldsymbol{x}^*,\boldsymbol{\eta}') - \bar{z}_i)] + \rho \sum_{i=1}^{k} (f_i(\boldsymbol{x}^*,\boldsymbol{\eta}') - \bar{z}_i)$. So $\max_{\boldsymbol{\xi}'\in\mathcal{U}} \max_i [w_i(f_i(\boldsymbol{x}',\boldsymbol{\xi}') - \bar{z}_i)] + \rho \sum_{i=1}^{k} (f_i(\boldsymbol{x}',\boldsymbol{\xi}') - \bar{z}_i) < \max_{\boldsymbol{\eta}'\in\mathcal{U}} \max_i [w_i(f_i(\boldsymbol{x}^*,\boldsymbol{\eta}') - \bar{z}_i)] + \rho \sum_{i=1}^{k} (f_i(\boldsymbol{x}^*,\boldsymbol{\eta}') - \bar{z}_i)$. This contradicts with the assumption that $\boldsymbol{x}^*$ is the optimal solution for (6). So $\boldsymbol{x}^*$ is a robust Pareto optimal solution for (3).

This result agrees with the sufficient condition presented in Theorem 4.4 in [2] for strongly increasing scalarizing functions, which states that the optimal solution of a strongly increasing scalarizing function is set-based minmax Pareto optimal to (3). In [2], the detailed proof was omitted. The necessary condition and the proof for strictly increasing scalarizing function are given in Theorem 4.1 in [2]. As a strongly increasing scalarizing function, (6) is also a strictly increasing scalarizing function. For the properties of strongly and strictly increasing scalarizing function see [2] and [20]. Based on (6), we introduce the robust ASF approach with (3) as the input to calculate a set of robust Pareto optimal solutions X_{rpo} as the output:

Step 1. Set $X_{rpo} = \emptyset$ and generate a set of reference points $\mathcal{Z}$.
Step 2. If $\mathcal{Z} = \emptyset$, stop.
Step 3. Choose a $\bar{\boldsymbol{z}} \in \mathcal{Z}$, and set $\mathcal{Z} = \mathcal{Z} \setminus \{\bar{\boldsymbol{z}}\}$.
Step 4. Find an optimal solution $\boldsymbol{x}^*$ to (6) using $\bar{\boldsymbol{z}}$ as the reference point and set $\boldsymbol{w}$ accordingly, e.g., $w_i = \frac{1}{z_i^{**} - \bar{z}_i}$, where $\boldsymbol{z}^{**}$ is the utopian objective vector. Set $X_{rpo} = X_{rpo} \cup \{\boldsymbol{x}^*\}$.
Step 5. Go to step 2.

In the robust ASF approach, we alter $\bar{\boldsymbol{z}}$ and set $\boldsymbol{w}$ accordingly for efficiently gaining a good representative set of robust Pareto optimal solutions X_{rpo}. When we evaluate their outcomes in the nominal case $\hat{\boldsymbol{\xi}}$, some of them can be dominated. We should only present nondominated solutions to the decision maker. So we refer to the robust Pareto optimal solutions whose corresponding outcomes are nondominated as nominal nondominated robust Pareto optimal solutions: a robust Pareto optimal solution $\boldsymbol{x}^*$ is a nominal nondominated robust Pareto optimal solution if there does not exist another $\boldsymbol{x} \in X_{rpo}$ such that $\boldsymbol{f}(\boldsymbol{x},\hat{\boldsymbol{\xi}}) \in \boldsymbol{f}(\boldsymbol{x}^*,\hat{\boldsymbol{\xi}}) - \mathbb{R}^k_{\geq}$.

For finding a nondominated robust Pareto optimal solution based on a NIMBUS classification, we solve a variant of the synchronous NIMBUS scalarizing problem presented in [13]:

$$
\begin{aligned}
\text{minimize} \quad & \max_{\substack{i\in I^{<} \\ j\in I^{\leq}}} [w_i(f_i(\boldsymbol{x},\hat{\boldsymbol{\xi}}) - z_i^*), w_j(f_j(\boldsymbol{x},\hat{\boldsymbol{\xi}}) - \hat{z}_j)] + \rho \sum_{i=1}^{k} w_i f_i(\boldsymbol{x},\hat{\boldsymbol{\xi}}) \\
\text{subject to} \quad & \boldsymbol{x} \in X_{rpo} \\
& f_i(\boldsymbol{x},\hat{\boldsymbol{\xi}}) \leq f_i(\boldsymbol{x}^c,\hat{\boldsymbol{\xi}}) \text{ for all } i \in I^{<} \cup I^{\leq} \cup I^{=}, \\
& f_i(\boldsymbol{x},\hat{\boldsymbol{\xi}}) \leq \epsilon_i \text{ for all } i \in I^{\geq},
\end{aligned}
\tag{7}
$$

where $I^{<}$, $I^{=}$, $I^{\geq}$, $I^{\leq}$, and $I^{\diamond}$ represent the corresponding classes of objectives and $\boldsymbol{x}^c$ is the current solution.

Proposition 1. *The solution of* (7) *is a nominal nondominated robust Pareto optimal solution for problem* (3).

Proof. Problem (7) is equivalent to a deterministic problem in the nominal case with the feasible set X_{rpo}. The proof that the solution of (7) is Pareto optimal for deterministic problems was given in [13]. Thus it fulfills the requirements to be a nominal nondominated robust Pareto optimal solution.

3.2 MuRO-NIMBUS

Based on the building blocks discussed above, we introduce MuRO-NIMBUS which can support the decision maker to find a most preferred solution for (3). We first discuss the idea of MuRO-NIMBUS in general. Then we present its steps followed by a discussion on the technical details of each step.

As mentioned before, e.g., in [1], the robustness and the quality of solutions usually conflict with each other. If the decision maker is not willing to sacrifice some quality to gain robustness, we can solve (3) in the nominal case as a deterministic problem. On the other hand, if the decision maker is willing to make some sacrifice to gain robustness, (s)he prefers to have a robust Pareto optimal solution by bearing the fact that its quality may not be as good as a Pareto optimal solution in the nominal case. MuRO-NIMBUS is developed for solving (3) when the decision maker is willing to sacrifice some quality to gain robustness. Because outcomes in the nominal case are very important for the decision maker and robustness of solutions can be guaranteed by finding best possible solutions in the worst case, we have three tasks during the solution process.

First, we need to guarantee the robustness of the solutions. Second, the nominal case has to be considered in terms of corresponding outcomes of solutions to satisfy the decision maker's preferences as much as can. Third, to help the decision maker to make an informed decision, corresponding outcomes in the worst case should be found. It is not possible to guarantee the robustness and consider two different kinds of realizations of the uncertain parameters at the same time during the solution process. So we separate the consideration into two stages in MuRO-NIMBUS: pre-decision making and decision making.

In the pre-decision making stage, we first concentrate on robustness, i.e., finding a set of robust Pareto optimal solutions. Then we consider the preferences of the decision maker in the decision making stage. Specifically, we support the decision maker to direct the solution process towards a most preferred robust Pareto optimal solution among the ones calculated. As a result, the final solution selected is robust Pareto optimal and at the same time corresponding to a most preferred outcome by the decision maker in the nominal case. In addition, the decision maker is informed of the outcome in the worst case.

We should be aware of the necessity of asking the decision maker whether (s)he is willing to sacrifice some quality to gain robustness before the solution process of a problem. Now we can present the overall algorithm of MuRO-NIMBUS as follows:

1. **Pre-decision making.**
   (a) Calculate the set X_{rpo} with the robust ASF approach. Calculate also the ideal and nadir objective vectors in the nominal case.
2. **Decision making**
   (a) Classify all the objectives into the class $I^{<}$ of the NIMBUS classification and solve (7) (by including only the first constraint) to find an initial nominal nondominated robust Pareto optimal solution $\boldsymbol{x}^c$.
   (b) Present the ideal and nadir objective vectors calculated in the nominal case to the decision maker.
   (c) Present the outcomes in the nominal and the worst cases corresponding to $\boldsymbol{x}^c$ to the decision maker. If the decision maker is satisfied, $\boldsymbol{x}^c$ is the final solution. Otherwise, continue.
   (d) Ask the decision maker to classify the objectives at the current solution, i.e., the outcome in the nominal case. Then solve (7) to find a new nominal nondominated solution and set it as $\boldsymbol{x}^c$ and go to step 2(c).

In step 1, the presence of the decision maker is not required. We use the robust ASF approach which can handle general problems (for example, the weighted-sum method in [5] assumes the problem to be solved is convex). In addition, in robust ASF, we apply the idea from [4] to alter the reference points $\bar{\boldsymbol{z}}$ and set $\boldsymbol{w}$ accordingly to efficiently obtain the set X_{rpo}. As for efficiently solving the scalarized problem and handling the constraints which should be fulfilled for all the possible realizations of the uncertain parameters, we discretize the uncertainty set to reformulate (6).

After step 1, we start the stage where the decision maker actively participates in the solution process. The goal is to find the most preferred solution from the set X_{rpo} by considering the corresponding outcomes in the nominal case. As an inherited advantage, MuRO-NIMBUS only requires the decision maker to classify the objectives based on the outcome of the current solution.

The decision making stage starts by calculating an initial nominal nondominated robust Pareto optimal solution. Before presenting the initial solution, the calculated ideal and nadir objective vectors are presented to the decision maker to help her/him to have a general idea on the ranges of the values of each objective function in the nominal case. With this information, when the outcome corresponding to the initial solution in the nominal case is presented, the decision maker can have a concrete understanding on its quality. As background information, the outcome(s) in the worst case is/are also shown to the decision maker to help her/him to make an informed decision.

As a tool for presenting the solutions to the decision maker, we utilize the value path visualization (see e.g., [6]). One can also modify some other visualization methods (see e.g., [12]) for this purpose. As said, depending on the characteristics of the involved uncertainty, there can exist multiple worst cases. We indicate the information on the outcomes in the worst cases accordingly in the visualization.

Figure 1 presents the idea of calculating the worst case objective function values in the visual presentation of a solution. In the figure, we have five different realizations of the uncertain parameters and the uncertain parameters do not relate to each other. The outcome in the nominal case is presented as the value path in the figure in blue. Outcomes with other realizations are presented in grey. By solving (4), we obtain the individual maxima of each objective in the uncertainty set as the outcome in the worst case. The corresponding outcome in the worst case is marked by triangles in the figure. The same idea applies when the uncertain parameters are related to each other. Instead of single values, we get ranges of values as the outcomes in the worst cases.

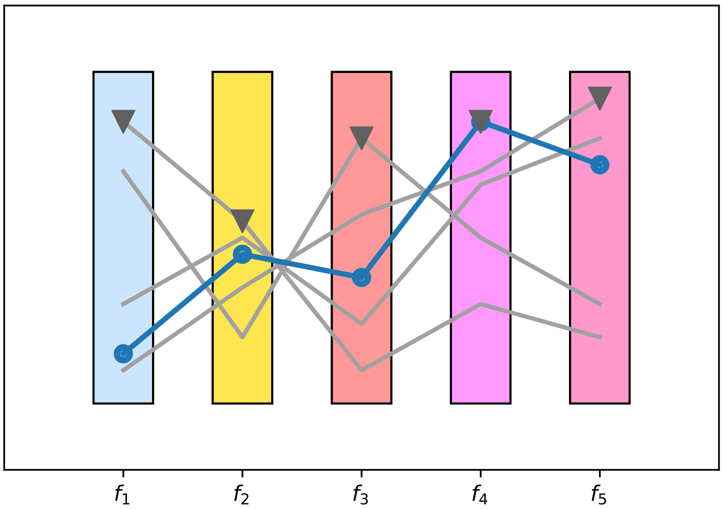

Fig. 1. Outcomes in the worst case

After having seen the initial solution, the decision maker can classify the objectives into up to five classes as discussed in Section 2 to express her/his preferences for a more desired solution. Based on the classification, we solve the scalarizing problem (7) to find a new nominal nondominated robust Pareto optimal solution which satisfies the classification best. The new solution is presented to the decision maker with an updated visualization. The solution process continues until the decision maker finds the most preferred nominal nondominated robust Pareto optimal solution.

4 Numerical Example

In this section, we simulate the solution process of the multiobjective ship design problem [21] to demonstrate the application of MuRO-NIMBUS. The problem has three objectives: minimizing the transportation cost, minimizing the light ship mass and maximizing the annual cargo. A detailed presentation of the problem in the deterministic case is in the Appendix A of [21].

The problem was originally studied as a deterministic problem. In the uncertain version studied in this paper, we consider two parameters which stem from given intervals: the fuel price and the round trip mileage. The fuel price affects the transportation cost. The round trip mileage affects both the transportation cost and the annual cargo. The fuel price can fluctuate for example due to the change of the energy market situation. The round trip mileage can vary if the weather conditions change. We treat the values of the two parameters in the deterministic formulation as their nominal values since they are supposed to describe the most typical values of the parameters. We implemented the problem in MATLAB®and used a build-in solver with MultiStart to find X_{rpo}.

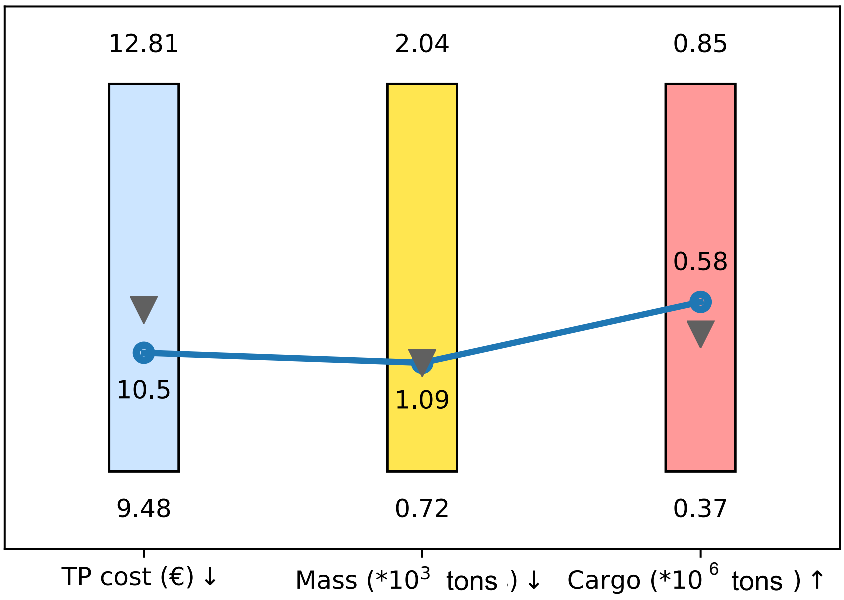

Fig. 2. Iteration 1 of ship design problem

Before the solution process, we communicated with the decision maker and she was willing to sacrifice some quality to gain robustness. In step 1 of MuRO-NIMBUS, we calculated a representative set of 150 robust Pareto optimal solutions with the robust ASF approach and we also calculated the ideal and nadir objective vectors in the nominal case. Based on our computational experiments, 150 solutions were sufficient for this problem. Then we started the first iteration of the decision making stage.

Step 2(a) We set the three objectives in $I^{<}$ and solved (7). We found an initial nominal nondominated Pareto optimal solution from X_{rpo}.

Step 2(b) We presented the ideal objective vector $\boldsymbol{z}^{*} = (9.479, 716.3, 0.8534)^T$ and the nadir objective vectors $\boldsymbol{z}^{nad} = (12.813, 2040.1, 0.372)^T$ in the nominal case to the decision maker. Their components corresponding to each objective are also shown in the visual illustration. In the visual presentation, we used 10^3 tonne as the unit, i.e., the ideal and nadir values for the light ship mass was marked as 0.7163 and 2.0401 respectively. To help the decision maker to quickly read the number, we used a million tonnes as the unit for annual cargo.

Step 2(c) Then we presented the initial outcome to the decision maker as illustrated in Figure 2. In the nominal case, 10.5 pounds/tonne for the transportation cost, 1090 tonnes light ship mass and the ship can handle 0.58 million tonnes cargo annually. The outcome in the worst case is marked in the figure. Even though one of the considered uncertain parameters affects two objectives, we had only one worst case because the two objectives are not conflicting with each other. The decision maker was not satisfied with the solution and wanted to continue the solution process.

Step 2(d) The decision maker specified her preferences by classifying the objectives and wanted to improve the annual cargo as much as she can while allowing the light ship mass to be impaired until 1800 tonnes. In the NIMBUS classification, this corresponds to: $I^{<} = \{f_3\}$, $I^{\geq} = \{f_2\}$ with $\epsilon_2 = 1800$ and $I^{\diamond} = f_1$. Based on this classification, we solved (7). As a result, we got a new nominal nondominated robust Pareto optimal solution.

Iteration 2. We presented the new solution to the decision maker as in Figure 3. The transportation cost was 9.51 pounds/tonne, and the light ship mass was 1640 tonnes while the annual cargo was 0.77 million tonnes in the nominal case. The decision maker observed in the visual presentation that the worst case outcome of the transportation cost did not degrade as much as in the initial outcome. Even though she seemed to have a solution whose outcome in the worst case did not degrade much compared to the outcome in the nominal case, she could not accept the light ship mass. So she decided to reduce the light ship mass to 1100 tonnes by allowing the transportation cost to increase until 10.9 pounds/tonne and the annual cargo to reduce until 0.5 million tonnes, i.e., she classified the objectives as $I^{\leq} = \{f_2\}$ with an aspiration level $\hat{z}_2 = 1100$ and $I^{\geq} = \{f_1, f_3\}$ with bounds $\epsilon_1 = 10.9$ and $\epsilon_3 = 0.5$. Based on this classification, problem (7) was solved to get a new solution.

Iteration 3. We presented the new solution to the decision maker as in Figure 4 with 10.59 pounds/tonne for the transportation cost, 1040 tonnes as the light ship mass and 0.57 million tonnes annual cargo. With this solution, the decision maker noticed that even though the light ship mass was quite low, the other two objectives were at the same time approaching her specified bounds. She also observed that the value of the first objective function has higher degradation than the previous solution. She understood that she cannot have lower light ship

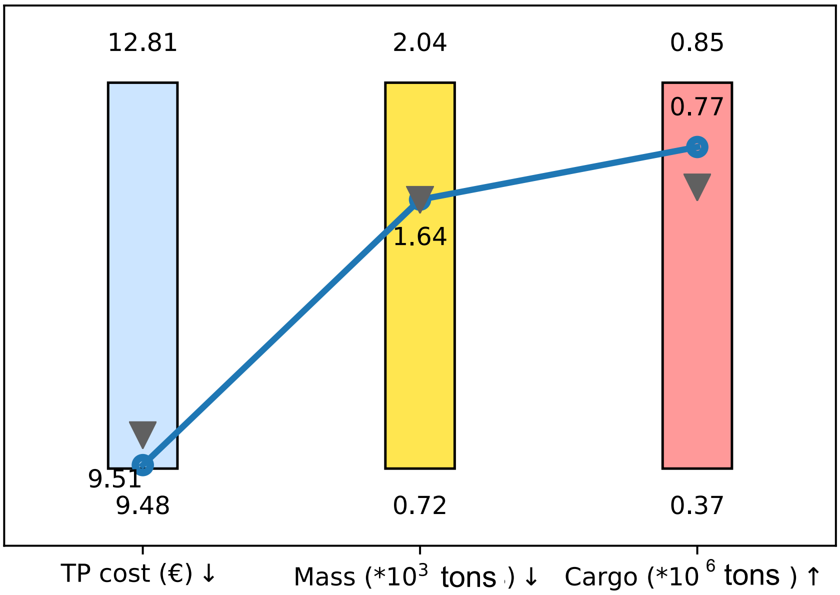

Fig. 3. Iteration 2 of ship design problem

mass if she is not willing to impair the other two objectives further and decided to stop. Naturally, if the decision maker is not satisfied, she can continue the solution process until she finds a most preferred solution.

During the solution process of the uncertain version of the multiobjective ship design problem, the decision maker was able to consider the outcomes in the nominal case with guaranteed robustness of solutions. Bearing in mind that the outcome in the nominal case of her final solution might not be as good as a deterministic Pareto optimal solution, she could still direct the interactive solution process towards a most preferred one among the robust Pareto optimal solutions according to their outcomes in the nominal case. Expressing her preferences by classifying the objectives did not bring her additional cognitive load. With the visualized information, she observed the outcomes of the solutions in the worst case in addition to the outcomes in the nominal case. Even though she could not interfere directly how the outcomes in the worst cases behaved, the information was critical for her to make an informed decision. In addition, if the worst case is realized, the solution the decision maker has would still be valid.

5 Conclusions

In this paper, we introduced MuRO-NIMBUS which is an interactive method for solving multiobjective optimization problems with uncertain parameters. In

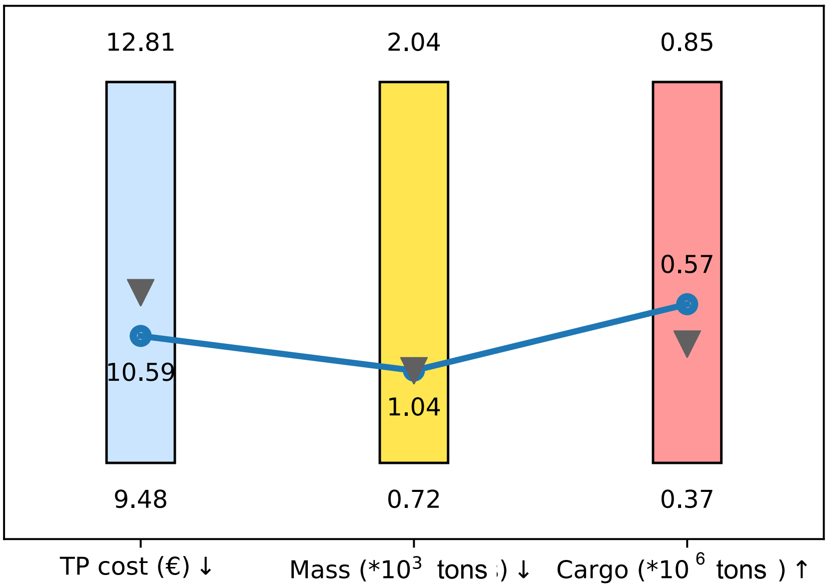

Fig. 4. Iteration 3 of ship design problem

MuRO-NIMBUS, we support the decision maker to find a preferred balance by interacting in the nominal case but also following what happens in the worst case. We divided the consideration of the robustness and the outcomes in the nominal cases into the pre-decision making and the decision making stages. With the two-stage solution process, the decision maker finds a robust Pareto optimal solution with a preferred outcome in the nominal case and at the same time, the outcome in the worst case is also acceptable. In this way, the information provided to and requested from the decision maker is understandable in MuRO-NIMBUS. The decision maker can also be easily involved in the interactive solution process without much additional cognitive load. By providing the information in both nominal and worst cases, MuRO-NIMBUS supports the decision maker to make an informed decision. We demonstrated the application of MuRO-NIMBUS with an example problem.

The development of MuRO-NIMBUS has initiated many avenues for further research. First, some additional features on the decision making stage can be developed. We can allow the decision maker to choose whether (s)he would like to find a most preferred solution based on the corresponding outcome in the nominal case (as is done in MuRO-NIMBUS), or in the worst case. This will allow the decision maker to consider different aspects during the decision making process. As an essential part to support the decision maker, we can also consider how to visualize the solutions more effectively. Second, a decision maker might want to find a robust Pareto optimal solution but with only a limited amount of sacrifice on the quality. To achieve this, we can study some other robustness concepts and analyze their properties from the decision making point of view aiming at finding a good trade-off between robustness and quality.

References

1. Bertsimas, D., Sim, M.: The price of robustness. Operations Research 52(1), 35–53 (2004)
2. Bokrantz, R., Fredriksson, A.: Necessary and sufficient conditions for Pareto efficiency in robust multiobjective optimization. European Journal of Operational Research 262(2), 682–692 (2017)
3. Branke, J., Deb, K., Miettinen, K., Slowinski, R. (eds.): Multiobjective Optimization, Interactive and Evolutionary Approaches. Springer (2008)
4. Cheng, R., Jin, Y., Olhofer, M., Sendhoff, B.: A reference vector guided evolutionary algorithm for many-objective optimization. IEEE Transactions on Evolutionary Computation 20(5), 773–791 (2016)
5. Ehrgott, M., Ide, J., Schöbel, A.: Minmax robustness for multi-objective optimization problems. European Journal of Operational Research 239(1), 17–31 (2014)
6. Geoffrion, A.M., Dyer, J.S., Feinberg, A.: An interactive approach for multicriterion optimization, with an application to the operation of an academic department. Management Science 19(4), 357–368 (1972)
7. Hassanzadeh, F., Nemati, H., Sun, M.: Robust optimization for multiobjective programming problems with imprecise information. Procedia Computer Science 17, 357–364 (2013)
8. Hassanzadeh, F., Nemati, H., Sun, M.: Robust optimization for interactive multiobjective programming with imprecise information applied to R&D project portfolio selection. European Journal of Operational Research 238(1), 41–53 (2014)
9. Ide, J., Schöbel, A.: Robustness for uncertain multi-objective optimization: a survey and analysis of different concepts. OR Spectrum 38(1), 235–271 (2016)
10. Kuhn, K., Raith, A., Schmidt, M., Schöbel, A.: Bi-objective robust optimisation. European Journal of Operational Research 252(2), 418–431 (2016)
11. Miettinen, K.: Nonlinear Multiobjective Optimization. Kluwer Academic Publishers (1999)
12. Miettinen, K.: Survey of methods to visualize alternatives in multiple criteria decision making problems. OR Spectrum 36(1), 3–37 (2014)
13. Miettinen, K., Mäkelä, M.M.: Synchronous approach in interactive multiobjective optimization. European Journal of Operational Research 170(3), 909–922 (2006)
14. Miettinen, K., Mustajoki, J., Stewart, T.J.: Interactive multiobjective optimization with NIMBUS for decision making under uncertainty. OR Spectrum 36(1), 39–56 (2014)
15. Sabioni, C.L., de Oliveira Ribeiro, M.F., de Vasconcelos, J.A.: Decision maker iterative-based framework for multiobjective robust optimization. Neuro computing 242, 113–130 (2017)
16. Sawaragi, Y., Nakayama, H., Tanino, T.: Theory of Multiobjective Optimization. Academic Press (1985)
17. Steuer, R.E., Choo, E.U.: An interactive weighted Tchebycheff procedure for multiple objective programming. Mathematical Programming 26(3), 326–344 (1983)

18. Steuer, R.: Multiple Criteria Optimization: Theory, Computation, and Applications. John Wiley & Sons, Inc. (1986)
19. Wiecek, M.M., Dranichak, G.M.: Robust multiobjective optimization for decision making under uncertainty and conflict. In: Gupta, A., Capponi, A., Smith, J.C., Greenberg, H.J. (eds.) Optimization Challenges in Complex, Networked and Risky Systems. pp. 84–114 (2016)
20. Wierzbicki, A.P.: On the completeness and constructiveness of parametric characterizations to vector optimization problems. OR Spectrum 8(2), 73–87 (1986)
21. Yang, J.B.: Minimax reference point approach and its application for multiobjective optimization. European Journal of Operational Research 126(3), 541–556 (2000)