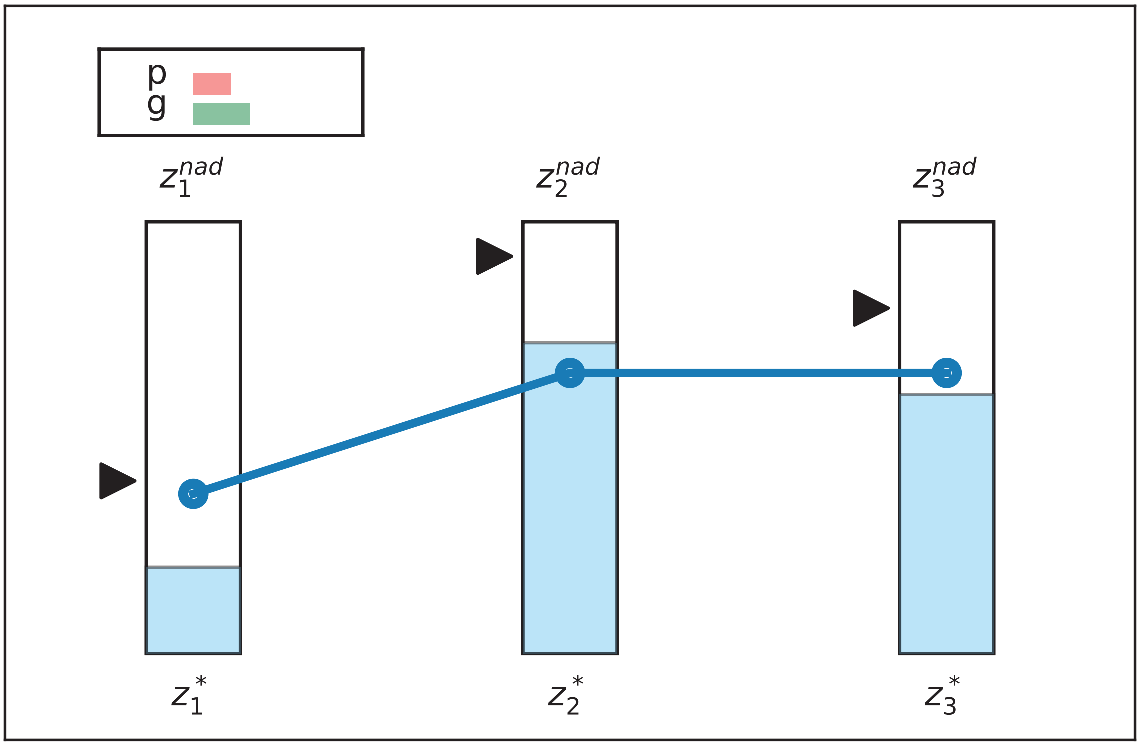
JYU DISSERTATIONS 16

Yue Zhou-Kangas

Interactive Methods for Multiobjective Robust Optimization



UNIVERSITY OF JYVÄSKYLÄ
FACULTY OF INFORMATION TECHNOLOGY

JYU DISSERTATIONS 16

Yue Zhou-Kangas

Interactive Methods for Multiobjective Robust Optimization

Esitetään Jyväskylän yliopiston informaatioteknologian tiedekunnan suostumuksella
julkisesti tarkastettavaksi yliopiston Agora-rakennuksen Lea Pulkkisen salissa
lokakuun 15. päivänä 2018 kello 12.

Academic dissertation to be publicly discussed, by permission of
the Faculty of Information Technology of the University of Jyväskylä,
in building Agora, Lea Pulkkinen hall, on October 15 2018 at 12 o'clock noon.

JYVÄSKYLÄN YLIOPISTO
UNIVERSITY OF JYVÄSKYLÄ

JYVÄSKYLÄ 2018

Editors
Timo Männikkö
Faculty of Information Technology, University of Jyväskylä
Ville Korkiakangas
Open Science Centre, University of Jyväskylä

Permanent link to this publication: http://urn.fi/URN:ISBN:978-951-39-7549-4

ISBN 978-951-39-7549-4 (PDF)
URN:ISBN:978-951-39-7549-4
ISSN 2489-9003

ABSTRACT

Zhou-Kangas, Yue
Interactive Methods for Multiobjective Robust Optimization
Jyväskylä: University of Jyväskylä, 2018, 78 p. (+included articles)
(JYU Dissertations
ISSN 2489-9003; 16)
ISBN 978-951-39-7549-4 (PDF)
Finnish summary
Diss.

Practical optimization problems usually have multiple objectives, and they also involve uncertainty from different sources. Various robustness concepts have been proposed to handle multiple objectives and the involved uncertainty simultaneously. However, the practical applicability of the proposed concepts in decision making has not been widely studied in the literature. Developing solution methods to support a decision maker to find a most preferred robust solution is an even more rarely studied topic. Thus, we focus on two goals in this thesis including 1) analyzing the practical applicability of different robustness concepts in decision making and 2) developing interactive methods for supporting decision makers to find most preferred robust solutions under different types of uncertainty.

We first consider decision uncertainty (i.e., the optimized solutions cannot be guaranteed with exact implementations). We propose a robustness measure to quantify the effects of uncertainty in the objective function values of solutions. We incorporate the robustness measure to an interactive method, where the solutions are presented to the decision maker with enhanced visualization.

We then consider parameter uncertainty (i.e., the parameters in the objective functions involve uncertainty). We first utilize the concept of set-based minmax robustness and develop a two-stage interactive method to support the decision maker to find a most preferred set-based minmax robust Pareto optimal solution. Since set-based minmax robust Pareto optimal solutions are difficult to compute, we propose an evolutionary multiobjective optimization method to approximate a set of them.

We then analyze different robustness concepts and verify that lightly robust Pareto optimal solutions are good trade-offs between robustness and objective function values. For supporting a decision maker to find a most preferred lightly robust Pareto optimal solution, we propose an interactive method. The results of this thesis extend the applicability of robustness concepts in decision making to practical problems. In addition, the proposed methods bring decision support in multiobjective robust optimization into practice.

Keywords: Robustness, multiobjective optimization, uncertainty, interactive methods, decision-making

Author Yue Zhou-Kangas
Faculty of Information Technology
University of Jyväskylä
Finland

Supervisor Professor Kaisa Miettinen
Faculty of Information Technology
University of Jyväskylä
Finland

Dr. Karthik Sindhya
Faculty of Information Technology
University of Jyväskylä
Finland

Reviewers Professor António Gaspar Lopes da Cunha
Department of Polymer Engineering
University of Minho
Portugal

Emeritus Professor Theodor Stewart
Department of Statistical Sciences
University of Cape Town
South Africa

Opponent Professor Margaret Wiecek
School of Mathematical and Statistical Sciences
Clemson University
USA

ACKNOWLEDGEMENTS

First, I would like to express my sincere gratitude to my supervisors Prof. Kaisa Miettinen and Dr. Karthik Sindhya for their support and guidance during my PhD. studies. Their expertise on multiobjective optimization and decision making under uncertainty has helped me in the process of maturing my research. I am eternally grateful for Prof. Kaisa Miettinen's openness to various new ideas and her patience in the process of preparing the thesis. I also especially appreciate Dr. Karthik Sindhya's encouragements in engaging new opportunities.

In addition to my supervisors, I would like to thank Prof. Dr. Anita Schöbel from the Institute of Applied Mathematics at the University of Göttingen for collaborations and insightful discussions and comments. I also would like to thank Prof. António Gaspar Lopes da Cunha from the University of Minho and Prof. Theodor Stewart from the University of Cape Town for reviewing my thesis. I especially express my gratitude to Prof. Stewart for providing many useful comments for improving the thesis from the initial draft.

My PhD. studies have been financially supported by COMAS Doctoral Program in Computing and Mathematical Sciences at the University of Jyväskylä, Faculty of Information Technology at the University of Jyväskylä, the vice rector research funding of Prof. Kaisa Miettinen, the mobility funding of the Science Council at the University of Jyäskylä, and Finnish University Partnership for International Development. My thesis was written under the thematic research area Decision Analytics utilizing Causal Models and Multiobjective Optimization (DEMO).

I would like to thank current and previous members of the Industrial optimization Group for the excellent research atmosphere. I especially would like to express my gratitude to Dr. Jussi Hakanen, Dr. Markus Hartikainen, Dr. Vesa Ojalehto, and Dr. Dmitry Podkopaev for their guidance with their experiences. I also especially appreciate the support in research from Dr. Mohammad Tabatabaei and Dr. Tinkle Chugh as well as the friendship offered by them.

During my research visit in Prof. Schöbel's research group, the members have warmly hosted me. I also thank all my friends for their friendship and supports. Last but not least, I would like to express my gratitude to my family for their love. My Parents Wu Aihua and Zhou Xuesheng have always encouraged me to pursue my dreams. I also thank my husband Mikko, daughter Olivia, and parents-in-law (Eeva and Matti) for their understanding and support.

LIST OF FIGURES

CONTENTS

LIST OF INCLUDED ARTICLES

PI Yue Zhou-Kangas, Kaisa Miettinen, and Karthik Sindhya. Solving Multiobjective Optimization Problems with Decision Uncertainty: an Interactive Approach. *Journal of Business Economics, to appear, doi: 10.1007/s11573-018-0900-1*, 2018.

PII Yue Zhou-Kangas, Kaisa Miettinen, and Karthik Sindhya. Interactive Multiobjective Robust Optimization with NIMBUS. *In: M. Baum, G. Brenner, J. Grabowski, T. Hanschke, S. Hartmann, A. Schöbel (eds) Simulation Science. SimScience 2017. Communications in Computer and Information Science, vol 889, p60-76, Springer.*, 2018.

PIII Yue Zhou-Kangas and Kaisa Miettinen. A Simple Indicator based Evolutionary Algorithm for Set-based Minmax robustness. *In A. Auger, C. M. Fonseca, N. Lourenco, P. Machado, L. Paquete, and D. Whitley (eds) Parallel Problem Solving from Nature - PPSN XV. PPSN 2018. Lecture Notes in Computer Science, vol 11101, p286-297, Springer.*, 2018.

PIV Yue Zhou-Kangas and Anita Schöbel. The Price of Multiobjective Robustness: Analyzing Solution Sets to Uncertain Multiobjective Optimization Problems. *Submitted to a journal*.

PV Yue Zhou-Kangas and Kaisa Miettinen. Decision Making in Multiobjective Optimization Problems under Uncertainty: Balancing Between Robustness and Quality. *Submitted to a journal*.

1 INTRODUCTION

Many optimization problems in practice have multiple objectives and involve uncertainty from different sources. These two issues complicate the process of finding solutions to the problems. Conventional multiobjective optimization methods often ignore uncertainty and consider multiple objectives simultaneously. However, it is also important to handle the uncertainty involved. For example, when designing a product, finding deterministic solutions can result in products with unexpected and undesired degradation in quality if the designed products can only be guaranteed to be produced with some tolerance interval. Also, when the input data of the problem is not known exactly or there are uncertain future developments, deterministic solutions can be rendered invalid.

Practitioners and researchers aim at finding solutions that are sufficiently immune to uncertainty involved while handling multiple objectives simultaneously. When the implementation of solutions cannot be guaranteed exactly, decision uncertainty is considered. It refers to the type of uncertainty which is reflected in the decision variables of the optimization problems. For example, decision uncertainty with multiple objectives is considered in [27] in designing an electromagnetic non-contact flow measurement device. In the device, the magnetic direction (a decision variable) cannot be implemented with the desired accuracy because the magnet direction can only be guaranteed with some tolerance interval. Considering only deterministic solutions can cause inaccurate measurements given by the devices.

When there is data that is unknown exactly, e.g., due to imprecise measurements or uncertain future developments, the uncertainty is usually reflected in parameters in the objective functions or constraints. For example, in the wood cutting problem considered in [45], the quality of wood can only be determined during the cutting process. Thus, the quality of the wood is treated as an uncertain parameter in the objective functions when generating a cutting plan. Without considering this uncertainty, implementing a cutting plan can result in e.g., using lower quality wood for higher quality orders. Other practical problems with parameter uncertainty include e.g., finding the shortest path in public transportations [52, 66], and optimizing investment portfolios [29, 39].

As mentioned before, conventional multiobjective optimization methods often ignore uncertainty and consider multiple conflicting objectives simultaneously to find a set of Pareto optimal solutions or support finding the most preferred Pareto optimal solution (see e.g., [18, 56, 70, 74]). Each solution corresponds to an objective vector which consists of the objective function values of the solution. We say that a solution is Pareto optimal if we cannot improve the corresponding value of an objective without impairing at least one of the others. We call the set of objective vectors corresponding to the Pareto optimal solutions the Pareto front. Usually, only one final solution is selected for implementation, and it is found by utilizing the preferences of a decision maker. A decision maker is a person who has expert knowledge in the problem domain. In this thesis, we assume that there is only one decision maker in each problem.

One can categorize multiobjective optimization methods into four classes for the decision maker to find a most preferred solution [42, 56]: no-preference methods, a priori methods, a posteriori methods, and interactive methods. When the decision maker does not have specific preferences about the final solution, we can find a neutral compromise solution (i.e., a solution corresponds to an objective vector somewhere in the "middle" of the Pareto front). In a priori methods, we first ask the decision maker to specify her or his preferences and find a solution which satisfies the preferences as well as possible. In a posteriori methods, we first calculate a (representative) set of Pareto optimal solutions and then the decision maker is expected to choose a final solution based on her or his preferences.

In interactive methods, we allow the decision maker to direct the solution process towards the most preferred solutions by iteratively specifying preferences. Different interactive methods have been successfully applied to support decision makers to find a most preferred solution for different types of problems (see e.g., [16, 56] for descriptions of different interactive methods and some applications). They provide the decision maker opportunities to learn about the problem and the feasible objective vectors. In addition, they also allow the decision maker to learn about her or his own preferences and how attainable they are.

As mentioned before, with uncertainty involved, Pareto optimal solutions can be rendered with unexpected and undesired degradations in their objective function values. Thus, considering uncertainty is as important as considering multiple objectives. The effects of uncertainty on the solutions can be studied via sensitivity analysis (see e.g., [69]) after the solutions have been found. There are also different approaches in the literature to consider objectives to be optimized and uncertainty simultaneously. The stochastic approach (see e.g., [13]) utilizes data to estimate a distribution of the involved uncertainty. The estimated distribution is utilized in reformulating the original objectives to e.g., consider the mean and variance of the objective function values. The reformulated objectives are then optimized. The fuzzy approach (see e.g., [46]) relies on a decision maker's expertise to judge the memberships of solutions under uncertainty. Robust optimization approaches (see e.g., [8, 10, 44, 79]) consider the possible realizations of uncertainty simultaneously and find solutions that are good with respect to them. In this thesis, we concentrate on robust optimization approaches and do not make any assumptions of sufficient data to estimate distributions nor rely on the expert judgment of a decision maker to build fuzzy membership functions.

The field of single-objective robust optimization (see e.g., [8, 10, 73]) has been studied since 1970s. The field of multiobjective robust optimization has only gained research attentions in recent years with a focus on the developments of robustness concepts to define Pareto optimal solutions under uncertainty (see summaries of the concepts in [44, 79]). It is hard to find discussions on the practical applicability of multiobjective robustness concepts in decision making. Also, very little attention has been paid to solution methods for supporting a decision maker to find a most preferred robust Pareto optimal solution.

However, as mentioned before, usually only one final solution is chosen to be implemented based on the decision maker's preferences. Thus, only defining and computing a set of robust Pareto optimal solutions is not enough for solving multiobjective optimization problems under uncertainty. We need to support the decision maker to find the final solution based on her or his preferences. Due to different possible realizations of uncertainty (which we call scenarios), the objective function values vary in different scenarios. A Pareto optimal solution in a scenario can be very bad in other scenarios. Since the existence of solutions which are good in every scenario can be rare, we need to help the decision maker to understand robustness (i.e., consequences of uncertainty) and at the same time to consider the preferences in objective function values. In addition, we also need to support the decision maker to understand and consider her or his preferences in the trade-off between robustness and objective function values (i.e., the quality of solutions).

In this thesis, we consider multiobjective robustness focusing on the practical applicability in decision making. We consider different robustness measures and analyze the minmax robust solutions [26, 29, 53], lightly robust solutions [44], and compare them with deterministic solutions. Minmax robustness optimizes in the worst case with respect to uncertainty. Light robustness finds the best solutions in the worst case under the condition that the degradation of their nominal objective function values are tolerable. The nominal case or scenario describes the most likely or undisturbed realization of uncertain parameters or variables. For finding deterministic solutions, we first identify a nominal case or scenario. Then, we solve the problem in the nominal case as a deterministic multiobjective optimization problem.

We also develop solution methods for supporting a decision maker to find a most preferred robust Pareto optimal solution. As mentioned before, interactive methods have many advantages, e.g., they enable the decision maker to learn about the problem, the feasible objective vectors, and her or his own preferences. Thus, the methods we develop in this thesis are interactive methods. We concentrate on decision uncertainty and parameter uncertainty in objective functions. The type of problems where uncertainty is reflected in the parameters in the constraints is not within the scope of this thesis since they can be handled in the same way as in single-objective optimization problems (as mentioned in [79]).

Supporting the decision maker to find a most preferred robust Pareto optimal solution is very challenging. The challenge comes from the simultaneous consideration of multiple objectives and uncertainty with the supports for the decision maker in the solution process. In this thesis, we tackle this challenge by considering the following four aspects:

1. As is usually the case, robustness and the objective function values corresponding to the solutions, which we call quality of the solutions, are conflicting with each other. It is desirable for the decision maker to quantify how much more "robust" the solutions can become by sacrificing some quality.
2. Robustness can have different meanings in different disciplines (for some examples, see [2, 41]). In addition, "robustness" does not have a natural meaning or unit, a clear definition of robustness with a practical meaning should be communicated to the decision maker so that (s)he knows what is to be expected and how to interpret the provided information.
3. During the solution process, the decision maker should be supported in terms of grasping a total balance not only on the multiple conflicting objectives but also on robustness.
4. During the interactive solution process, the decision maker should not be exposed to a too heavy cognitive load. The information shown to the decision maker and the information requested from the decision maker should be carefully considered.

With the collection of five papers [PI, PII, PIII, PIV, PV], we address the four aspects introduced above and develop interactive methods to support decision makers to find a most preferred robust Pareto optimal solutions. Paper [PI] concentrates on decision uncertainty and proposes a robustness measure to quantify the effects of decision uncertainty on solutions. We visualize the solutions to help the decision maker to understand the behaviors of solutions under decision uncertainty. An interactive method with the new robustness measure incorporated is also proposed to support the decision maker to find a most preferred solution. As a result, the decision maker is supported in being aware of and able to accept the behaviors of the solutions when the solutions cannot be implemented exactly.

Paper [PII] focuses on parameter uncertainty. In the paper, we propose the MuRO-NIMBUS method to support the decision maker to find the most preferred robust Pareto optimal solution. In MuRO-NIMBUS, we employ the concept of set-based minmax robustness [26] and introduce a robust version of an achievement scalarizing function [81] approach to find a set of set-based minmax robust Pareto optimal solutions. We then interact with the decision maker and visualize the solutions. With the two-stage process, the decision maker can be supported to find a most preferred set-based minmax robust Pareto optimal solution based on their objective function values in the nominal case. The final solution satisfies the decision maker's preferences best and it is at the same time valid even when the worst case happens.

In the literature, clear instructions on how to compute minmax robust Pareto optimal solutions are not always given. We can solve the robust version of the scalarized subproblem in MuRO-NIMBUS by e.g., approximation. For finding a set of set-based minmax robust Pareto optimal solutions, we need to solve multiple subproblems. In order to ease the computation, we develop a simple indicator-based evolutionary algorithm for set-based minmax robustness (SIBEA-R) in Paper [PIII] to approximate a set of set-based minmax robust solutions. The approximated set of solutions can then be used in MuRO-NIMNBUS when interacting with the decision maker.

As mentioned before, one of the goals of the thesis is to analyze the practical applicability of different robustness concepts in decision making. The conservativeness of minmax robustness has been recognized in single-objective cases (e.g., in [12]). By conservativeness, we mean that the objective function values of minmax robust solutions can be very bad in other scenarios even though they are the best in the worst case. In order to facilitate the development of supporting a decision maker to find a less conservative solution (than minmax robust solutions), we analyze the relationships of three different kinds of robust solutions and prove that lightly robust solutions are good trade-offs between robustness and quality in Paper [PIV]. Based on our findings in the analysis, we also propose two strategies to support decision making.

After the finding that lightly robust solutions are good trade-offs between robustness and quality, we develop an interactive method LiRoMo in Paper [PV] to support the decision maker to find a most preferred lightly robust Pareto optimal solution. In this method, we propose a reformulation of the scalarized lightly robust subproblem under some assumptions to efficiently calculate lightly robust Pareto optimal solutions. We also visually support the decision maker to understand how much objective function values are sacrificed to gain robustness in a solution and how much a nominal solution lacks of robustness.

The rest of the thesis is organized as follows. In Chapter 2, we present background information and basic notations and definitions related to multiobjective optimization. In Chapter 3, we introduce notations related to multiobjective optimization problems under uncertainty, review the related literature of multiobjective robust optimization and describe the fundamental robustness concepts relevant for this thesis. We summarize the proposed interactive method for solving multiobjective optimization problems under decision uncertainty in Chapter 4 followed by the summary of the MuRO-NIMBUS in Chapter 5. The analysis of the relationships between three different kinds of robust solutions is summarized in Chapter 6 together with the two proposed decision support strategies. The LiRoMo method is presented in Chapter 7. After summarizing the research done for this thesis, some discussions on related issues are presented in Chapter 8. The author's own contribution is described in Chapter 9 followed by the conclusions and future research directions in Chapter 10.

2 MULTIOBJECTIVE OPTIMIZATION

2.1 Multiobjective Optimization Problems

We consider multiobjective optimization problems in the following form:

$$\begin{array}{ll} \text{minimize} & f(x) = (f_1(x), ..., f_k(x))^T \\ \text{subject to} & x \in X. \end{array} \tag{1}$$

In the formulation of (1), $f_i(x), i = 1, ..., k$ are called objective functions, which are the components of the k-dimensional objective vector $f(x)$. The decision vectors x consist of the decision variables $(x_1, ..., x_n)^T$ which belong to a feasible set $X \subseteq \mathbb{R}^n$. In some of the included papers (e.g., in [PIV]), we use $\mathfrak{X}$ to represent the feasible set. The corresponding set in the objective space mapped by the objective functions $Z = f(X)$ is called the feasible objective set. We refer to the objective function values, i.e., the objective vector, of a solution as the outcome or the quality of the solution. When we compare two feasible solutions, we use the concept of dominance which is defined as follows:

Definition 1. *Let $x^1, x^2 \in X$, x^1 dominates x^2 if $f_i(x^1) \leq f_i(x^2)$ for all $i = 1, ..., k$ and $f_j(x^1) < f_j(x^2)$ in at least one index j.*

For (1), there usually exists a set of mathematically equally good Pareto optimal solutions defined as follows:

Definition 2. *A feasible solution $x^* \in X$ is Pareto optimal, if there does not exist $x \in X$ and $x \neq x^*$ such that $f_i(x) \leq f_i(x^*), i = 1, ..., k$ and for at least one index j, it holds that $f_j(x) < f_j(x^*)$.*

The set of Pareto optimal solutions is called the Pareto optimal set. We say that an objective vector is Pareto optimal if its corresponding decision vector is Pareto optimal. The set of Pareto optimal objective vectors is called the Pareto front.

We can also define Pareto optimal solutions with the help a non-negative ordering cone $\mathbb{R}^k_{\geq} = \{y \in \mathbb{R}^k : y_i \geq 0 \text{ for all } i = 1, ..., k\}$. We say that a feasible solution $x^* \in X$ is Pareto optimal if there does not exist $x \in X$ and $x \neq x^*$, such that $f(x) \in f(x^*) - \mathbb{R}^k_{\geq}$. In the literature, Pareto optimal solutions are also called efficient solutions. In the papers included in this thesis, we use both terms as synonyms.

For decision making, it is often useful for the decision maker to know the ranges of the Pareto front. The information of the ranges can be provided by the ideal objective vector z^{ideal} and the nadir objective vector z^{nad}. The ideal objective vector consists of the individual minima of the objective functions. The nadir objective vector provides the upper bounds on the values of the objectives in the Pareto front. It can be approximated by for example a so-called payoff table [56, 74]. The approximation can over or under estimate the nadir objective vector. Other methods to compute the nadir objective vector have been presented in e.g., [23]. For computational reasons, we also have the utopian objective vector $z^{\text{uto}} = (z_1^{\text{ideal}} - a, ..., z_k^{\text{ideal}} - a)^T$, where $a > 0$ is a small scalar. For simplicity, we assume that the objective functions are to be minimized in the formulation. Objective functions to be maximized can be easily converted to be minimized by multiplying by -1.

In order to solve a multiobjective optimization problem (1), a common technique is scalarization. Scalarization means to transform a multiobjective optimization problem to a single objective optimization subproblem such that its optimal solution is a Pareto optimal solution to (1). For solving the scalarized subproblem, a proper single-objective optimization method should be used. Achievement scalarizing function (see e.g., [80, 81]) is one of the widely used scalarizing function and it has different variants. Their characteristics are summarized in [80]:

Definition 3. *Let us consider $x^1, x^2 \in \mathbb{R}^n$,*

(i) A function g is strictly increasing, if $x_i^1 < x_i^2$ for all $i = 1, ..., n$ implies $g(x^1) < g(x^2)$.
(ii) A function g is strongly increasing, if $x_i^1 \leq x_i^2$ for all $i = 1, ..., n$ and $x_j^1 < x_j^2$ imply $g(x^1) \leq g(x^2)$.

Based on the definition, a strongly increasing achievement scalarizing function is also a strictly achievement scalarizing function. We utilize the following subproblem based on a variant of an achievement scalarizing function:

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

where ρ is a small scalar. The trade-off, which is bounded by ρ and $1/\rho$, represents the ratio of change in the objective function values when one of the objective increases while some others decrease. The solutions found by solving (2) are also called properly Pareto optimal solutions (see, e.g., [81]). For simplicity, we call them Pareto optimal solutions. The reference point is represented by $\bar{z}$. The 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 w sets a direction with which the reference point is projected onto the Pareto front.

Subproblem (2) can also be reformulated to a differentiable from (assuming that the objective functions are differentiable):

$$\begin{array}{ll} \text{minimize} & \alpha + \rho \sum_{i=1}^{k} w_i(f_i(x) - \bar{z}_i) \\ \text{subject to} & w_i(f_i(x) - \bar{z}_i) \leq \alpha \text{ for all } i = 1, ..., k \\ & x \in X, \end{array} \tag{3}$$

where the auxiliary scalar variable α is used for the transformation from the min-max form (2).

The achievement scalarizing function has many advantages. As discussed in the literature (e.g., [16, 56, 81]), an optimal solution of (2) is a Pareto optimal solution for (1) and any Pareto optimal solution with trade-offs bounded by ρ and $1/\rho$ can be found by changing $\bar{z}$. The reference point can be feasible or infeasible and the problem can be convex or nonconvex.

2.2 Interactive Multiobjective Optimization Methods

As mentioned before, interactive methods allow the decision maker to direct the solution process to find a most preferred Pareto optimal solution. Figure 1 presents the basic steps of a typical interactive method. It starts with presenting an initial solution to the decision maker (marked as DM in the figure). We call the presented solution which the decision maker is expected to consider the current solution x^c. The decision maker can study the current solution and consider if (s)he is satisfied. If the decision maker is satisfied, (s)he takes the current solution as the final solution and we terminate the solution process. If the decision maker is not satisfied, (s)he can express preferences to get a more desirable solution. Typically, preferences are incorporated into a scalarization function to calculate a new solution which satisfies the preferences as well as possible. This process continues until the decision maker finds a most preferred solution.

According to [16], based on the type of preference information the decision maker is expected to provide, interactive methods can be categorized into reference point based methods (see e.g., [51, 82]), classification based methods (see e.g., [58, 60, 70]) and trade-off based methods (see e.g., [18, 31]). In this thesis, the LiRoMo method proposed in [PV] is a reference point based method and the methods proposed in [PI, PII] are extensions of the NIMBUS method [58, 60], which is a classification based method.

In reference point based methods, the decision maker specifies her or his preferences as aspiration levels, which are the components of a reference point. The aspiration levels are the desired values of the objective functions. A Pareto optimal solution is found by solving a subproblem based on an achievement scalarizing function (e.g., (2) or (3)) by projecting the reference point with the

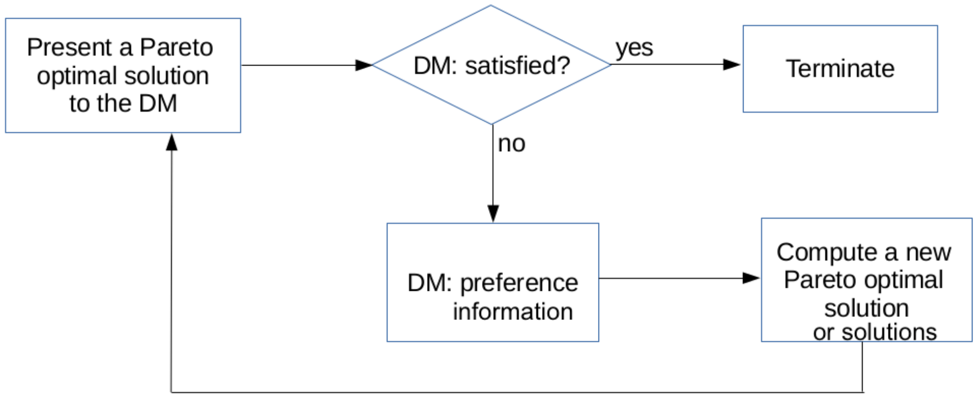


FIGURE 1 Basic steps of an interactive method

projection direction w onto the Pareto front. After having been shown a solution, the decision maker can consider whether it is satisfactory or not. If not, (s)he can provide a new reference point. Based on the new reference point, a new Pareto optimal solution can be found by solving the subproblem. The solution process continues until the decision maker finds a most preferred Pareto optimal solution.

NIMBUS has different versions depending on the scalarized subproblems used (see e.g., [58, 59]). In NIMBUS, the decision maker is expected to classify the objectives. The classification indicates what kind of solution would be more preferred than the current one. The objectives can be classified into up to five different classes including:

$I^{<}$ for those to be improved,
$I^{\leq}$ for those to be improved until some desired aspiration level $\bar{z}_i$,
$I^{=}$ 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. The aspiration levels should be better than the current objective function values and the bounds should be higher than the current objective function values. If the classification is feasible, i.e., the decision maker allows at least one of the objectives to be impaired to improve some other objectives, a scalarized subproblem is solved to find a new Pareto optimal solution reflecting the preferences.

In [PI, PII], we utilize a subproblem from synchronous NIMBUS [60]:

$$\begin{aligned} \text{minimize} \quad & \max_{\substack{i \in I^{<} \\ j \in I^{\leq}}} [w_i(f_i(x) - z_i^{\text{ideal}}),\ w_j(f_j(x) - \bar{z}_j)] + \rho \sum_{i=1}^{k} w_i f_i(x) \\ \text{subject to} \quad & x \in X \\ & f_i(x) \leq f_i(x^c) \text{ for all } i \in I^{<} \cup I^{\leq} \cup I^{=}, \\ & f_i(x) \leq \epsilon_i \text{ for all } i \in I^{\geq}, \end{aligned} \tag{4}$$

where $I^{<}$, $I^{=}$, $I^{\geq}$, $I^{\leq}$, and $I^{\diamond}$ represent the corresponding classes of objectives and x^c is the current solution. The synchronous NIMBUS method can find up to four solutions which satisfies the decision maker's preferences best in each iteration. In this thesis, we consider only one solution found by solving (4).

2.3 Evolutionary Multiobjective Optimization Methods

In addition to using scalarization techniques to find Pareto optimal solutions, we can also use evolutionary multiobjective optimization methods to approximate a set of Pareto optimal solutions. Figure 2 illustrates the basic steps of a typical evolutionary multiobjective optimization (EMO) method.

As illustrated in the figure, the method starts with a random initial population which consists of a set of solutions called individuals. We call the initial population the first generation. Then the individuals are evaluated to determine their objective function values. Based on the objective function values, environmental selection process chooses the "better" (e.g., in the sense of dominance) individuals with a larger probability to fill an intermediate mating pool. Based on the individuals in the mating pool, new individuals which are called offspring are generated by variation operators such as crossover and mutation. In the elitism step, the offspring are combined with their parents to guarantee that good individuals are taken to the next generation. The "better" individuals in the combined population are selected to form a new generation of population. The solution process continues until a termination criterion is met. As in the figure, the termination criterion is usually the maximum number of generations. After the termination, the final population is the output. For more details on evolutionary multiobjective optimization methods see e.g., [16, 20, 21].

The final population obtained by applying an evolutionary multiobjective optimization method is a set of non-dominated solutions. There is no guarantee in general that they are Pareto optimal solutions. In some literature, non-dominated solutions and Pareto optimal solutions are used as synonyms. In this thesis, they carry different meanings. We refer to the solutions obtained by methods which have theoretical proofs (e.g., by solving subproblem (2)) as Pareto optimal solutions. We refer to the final population given by an evolutionary multiobjective optimization method as non-dominated solutions. In addition, a set of individuals in a generation which do not dominate each other is also called non-dominated solutions.

The simple indicator-based evolutionary algorithm (SIBEA) [87] is an example of evolutionary algorithms. We introduce SIBEA here because we propose a variant of it for finding non-dominated set-based minmax robust solutions in this thesis. The SIBEA method access the domination count of individuals (i,.e., the number of other individuals a individual dominates). Based on the domination count, the individuals are ranked with non-dominated sorting. Non-dominated sorting is a procedure for ranking the individuals into different fronts. The rank is

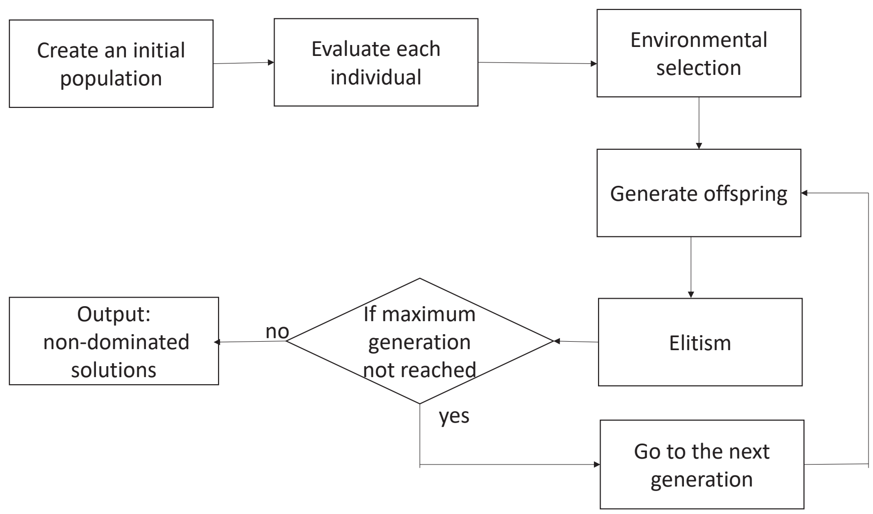


FIGURE 2 Basic steps of an EMO method

combined with an indicator function to assign fitness of the individual, which describes how good the individual is. Based on the rank and the indicator function, SIBEA selects "better" individuals into the next generation.

The hypervolume indicator (see e.g., [88, 90]) can be used in the SIBEA method as the indicator function. It is based on the volumes of the hypercubes formulated by the individuals and a reference point. Note that the reference point here is not the same as the reference point in (2). The reference point here can be formed by e.g., the worst values of each objective function in each generation. The formal definition of hypervolume indicator is as follows:

Definition 4. *The hypervolume indicator of a set of non-dominated objective vectors A is defined via the attainment function $\alpha_A(z)$ of objective vectors z as*

$$I_H(A) = \int_z \alpha_A(z)dz.$$

The SIBEA method takes the population size NP, the number of generations NG, and an indicator function (e.g., the hypervolume indicator) as inputs and produces a set of non-dominated solutions A as the output. The basic steps of the SIBEA method are as follows:

Step 1 (Initialization) Generate an initial population P and set the generation counter $m = 1$.

Step 2 (Mating) Generate an offspring population Q using crossover and mutation and set $P = P \cup Q$.

Step 3 (Environmental selection) Rank the population P using non-dominated sorting and identify different fronts $F^i, i = 1, 2, ...$ Set a new population $P' = \emptyset$. Then do the following:

(a) Set a counter $i = 1$ and set $P' = P' \cup F^i$ as long as $|P'| \leq NP$ and set $i = i + 1$, where $|P'|$ is the cardinality of P'.

(b) If $|P'| = NP$, set $P = P'$ and go to step 4. Otherwise determine the set of individuals P'' in P' with the worst rank.

(c) For each individual in P'', determine the loss of the hypervolume $I_H(f(P'')) - I_H(f(P'' \setminus \{x\}))$ if the individual x is removed from P''.

(d) Remove the individual with the smallest loss until $|P'| = NP$ and set $P = P'$.

Step 4 (Termination) If $m \geq NG$, terminate and set $P = A$. Otherwise set $m = m + 1$ and go to Step 2.

3 MULTIOBJECTIVE OPTIMIZATION UNDER UNCERTAINTY

3.1 Multiobjective Optimization Problems under Uncertainty

The multiobjective optimization problem presented in (1) is a deterministic formulation. When uncertainty is taken into account, the formulation of the problems changes. In this thesis, decision uncertainty and parameter uncertainty in the objective functions are considered. For decision uncertainty, we consider the following formulation:

$$\begin{array}{ll} \text{minimize} & f(x+\Delta x) = (f_1(x+\Delta x), ..., f_k(x+\Delta x))^T \\ \text{subject to} & x+\Delta x \in X \text{ for all } \Delta x \in \mathcal{U}, \end{array} \tag{5}$$

where Δx represent the unknown possible perturbation of the decision vectors and $\mathcal{U}$ is a hyperbox in the neighborhood of a nominal solution x^b, i.e., a computed deterministic solution without any perturbation. This solution can also be called the base solution. For solving problem (5), we consider all possible values of Δx and find nominal solutions with respect to both its nominal objective function values $f(x^b)$ and the perturbed objective function values $f(x^b + \Delta x)$.

For parameter uncertainty, we consider the following formulation:

$$\left(\begin{array}{ll} \text{minimize} & f(x,\xi) = (f_1(x,\xi), ..., f_k(x,\xi))^T \\ \text{subject to} & x \in X \end{array} \right)_{\xi \in \mathcal{U}}, \tag{6}$$

where ξ is a vector consisting of the uncertain parameters and $\mathcal{U}$ represents the uncertainty set where the uncertain parameters stem from. Each particular realization of $\xi \in \mathcal{U}$ is called a scenario and it corresponds to a deterministic multiobjective optimization problem. In this setting, problem (6) can be interpreted as a collection of deterministic multiobjective optimization problems. We refer to the set of objective vectors as the outcome set of a solution x for all $\xi \in \mathcal{U}$, which is denoted by $f_{\mathcal{U}}(x) = \{f(x,\xi) : \xi \in \mathcal{U}\}$.

As mentioned before, in robust optimization approaches, the uncertain parameters are assumed to stem from an uncertainty set $\mathcal{U}$. In the literature, different uncertainty sets are considered. In this thesis, we consider interval uncertainty and polyhedral uncertainty. Interval uncertainty is defined as follows:

$$\mathcal{U} = \{\xi_j \in [\underline{\xi}_j, \overline{\xi}_j] \text{ for } j = 1, ..., m\}$$

where $\underline{\xi}_j$ is the lower bound of the j-th uncertain parameter ξ_j and $\overline{\xi}_j$ is its upper bound. In this setting, there are m uncertain parameters. We assume that the nominal values of the uncertain parameters are in the interval $\hat{\xi}_j \in [\underline{\xi}_j, \overline{\xi}_j]$ for all $j = 1, ..., m$. Polyhedral uncertainty is defined by a set of m scenarios $\xi^1, ..., \xi^m$ as the extreme points of the convex hull:

$$\mathcal{U} = conv\{\xi^1, ..., \xi^m\}.$$

3.2 Robustness in Multiobjective Optimization

In the literature, different ways of handling uncertainty have been proposed to find solutions for (5) and (6). In order to consider uncertainty and multiple objectives simultaneously, there are two main types of approaches proposed. The first type is to use a robustness measure to quantify the changes of objective function values of the solutions due to uncertainty. The measures are usually used to quantify how "robust" a solution is. The second type is to alter the definition of Pareto optimality for concepts of robust Pareto optimality. We can search for robust Pareto optimal solutions by utilizing these concepts. Robustness concepts usually do not quantify how "robust" a solution is. In this section, we discuss the existing methods and approaches in the literature.

3.2.1 Robustness Measures

The definitions of robustness measures are usually based on the neighborhood of a nominal solution. Among the robustness measures, there are two different ways to quantify the robustness of the nominal solutions. One is to quantify the changes of the objective function values due to perturbations of solutions. The other is to quantify how much a solution can change with respect to a predefined tolerance in the objective function values.

In [22], a robustness measure is defined to compare the objective function values of a nominal solution and the average objective function values of sampled solutions in its neighborhood. In [30], the average difference between the objective function values of a nominal solution and those of the sampled solutions in the neighborhood is used to measure the robustness of the nominal solution. In [68], the robustness of a nominal solution is the largest difference between the worst and the best objective function values in the neighborhood.

Another common way is to study the mean and variance of the objective function values corresponding to the sampled solutions in the neighborhood as proposed in e.g., [3, 72]. For simulation-based optimization problems, representative models of the mean and variance of the objective function values can be built with surrogate models as in e.g., [55, 76]. Note that these research are not stochastic approaches since the computation of mean and variance is based on samples, not probability distributions.

In addition, there are also approaches which combine the stochastic approach and robustness measures e.g., in [37, 48]. In these approaches, when there is enought data to estimate probability distributions of uncertain parameters, the uncertain parameters are transformed to stochastic variables. When it is not possible to estimate the probability distribution, the objective functions in the neighborhood of nominal solutions are used to quantify the robustness.

As mentioned before, the robustness of a nominal solution can also be quantified by measuring how much the solution can change based on some predefined tolerance on the objective function values. In [36, 54], the maximum change of the nominal solution such that the objective function values are within tolerable values is used to quantify robustness. In [6], the percentage of solutions in the neighborhood whose objective function values are within the tolerance is used to quantify the robustness of a nominal solution. Alternatively, the stability of nominal Pareto optimal solutions are also studied (e.g., in [32, 85]). This line of research studies the conditions under which a nominal Pareto optimal solution remain Pareto optimal when objective functions change.

In order to consider robustness and the objective function values simultaneously, the robustness measures can be incorporated in difference ways. They can be used to replace the original objective function values (in e.g., [22, 55, 76]) or combined together with the original objective functions (in e.g., [3, 30, 72]). They can alternatively be used as additional constraints in the problem formulation (in e.g., [22, 54]). They can also be used to compare a set of solutions in an evolutionary multiobjective optimization method as in [6, 68].

From the decision maker's point of view, the existing measures are difficult to understand. The decision maker can only rely on the intuition if the value of a measure is the smaller the better or the bigger the better. The robustness measure proposed in [PI] is based on the comparison of the objective function values of the nominal solution and those of the perturbed solutions in its neighborhood. The comparison is linked with the ideal and nadir objective function values to provide understandable information for the decision maker on how uncertainty affects the objective function values of the nominal solution. For simultaneously consideration of robustness and nominal objective function values, we use the robustness measure as an additional objective to introduce trade-off.

3.2.2 Robustness Concepts

In addition to the robustness measures, the definition of Pareto optimality has been altered for defining robust Pareto optimality. The most widely studied robustness concepts belong to the family of minmax robustness. The idea of minmax robustness is to optimize in the worst case. Point-based minmax robustness [29, 53] uses the worst value of each objective function as the worst case. Set-based minmax robustness, which was originally proposed in [26] and generalized in [43], treats the worst case with respect to multiple objectives, which result in a set of worst case scenarios. Set-based dominance from set-valued optimization [49] is used to refine the dominance relationships of two feasible solutions based on their outcome sets.

Hull-based minmax robustness [15] forms a convex hull with the set of worst case scenarios of each solution and compares the convex hulls with the help of an ordering cone. When the problem is objective-wise uncertain (i.e., the uncertain parameters in the objectives do not relate to each other, see [26] for a formal definition), point-based minmax robustness, set-based minmax robustness, and hull-based robustness coincide with each other. The concept of minmax robustness has been applied in a wood cutting problem [45], and a portfolio optimization problem [29].

In the literature, it is not always clear how to compute minmax robust Pareto optimal solutions. For set-based minmax robust solutions, the authors of [26] proposed weighted-sum and epsilon-constraint approaches. Reformulations for efficient computation of point-based minmax robust solutions are proposed in [29, 33, 53] under assumptions on the characteristics of the problems. The idea of comparing the set of outcomes of each solution is also utilized in [4, 50], where the uncertainty set consists of a set of discrete scenarios. The comparison of the sets is embedded in an evolutionary algorithm to compute a set of non-dominated solutions. The SIBEA-R method proposed in [PIII] takes this kind of approach in an evolutionary algorithm.

As mentioned before, the conservativeness of minmax robustness is recognized in the literature. There have been also attempts to extend the idea of finding trade-offs between robustness and quality of solutions in single-objective robust optimization [7, 9, 12, 28, 35, 71]. Recently, similar line of research has been done in multiobjective settings. In [44, 52], the concept of light robustness [28, 71] is extended to multiobjective optimization problem under uncertainty. Light robustness aims at optimizing in worst case among those solutions whose nominal objective function values are tolerable. The concept of light robustness is applied in a shortest path problem [52], where the solution method is for combinatorial problems with uncertain parameters in one of the objectives in bi-objective optimization problems.

Other less-conservative multiobjective robustness concepts include for example the extensions of considering cardinality-constrained uncertainty based on [11, 12] into multiobjective settings. In [38, 39], the extension is for portfolio optimization problems with uncertain data. The idea of this kind of consideration is based on the thinking that not all uncertain parameters attain their worst case at the same time. This concept is to consider only a predefined maximum number of uncertain parameters for the worst case. The solution method proposed in [38, 39] is for linear problems with uncertain parameters stemming from an interval. In [66, 67], the consideration of this kind of uncertainty is extended to multiobjective combinatorial optimization problems. The authors also proposed two solution approaches with applications to shortest path problems under uncertainty for hazardous material transportation.

Another kind of concept is to optimize in the worst case but search for solutions whose objective function values in a specific scenario is as "near" as possible to a Pareto optimal objective vector in that scenario. Regret robustness takes this kind of approach. The idea is to find a minmax robust solution such that it deviates the least when compared to a Pareto optimal solution in a specific scenario. Different variations of multiobjective regret robustness have been proposed for different specific application problems in [63, 83]. In [83], regret robustness from single-objective optimization has been applied to scalarized subproblems. In [63], regret robustness is defined for multiple objectives in binary optimization problems.

In addition, there are also other concepts of multiobjective robustness. The concept of highly robust Pareto optimality is originally proposed in [14] for multiobjective linear optimization problems with interval uncertainty. The idea is to find solutions which are Pareto optimal in all possible scenarios. This concept is generalized to other types of uncertainty sets in [52] and to general multiobjective optimization problems in [44]. Recently, the conditions of the existence of highly robust Pareto optimal solutions in multiobjective convex optimization problems are presented in [34]. In addition, the characteristics and properties of highly robust Pareto optimal solutions in multiobjective linear optimization problems are studied in [25]. The authors also derived the lower and upper bounds of the highly robust Pareto optimal solutions in this class of problems. As can be observed in the literature which studies the conditions of existence of highly robust Pareto optimal solutions, it can be the case that highly robust Pareto optimal solutions do not exist for some problems. Even though this kind of solutions can be desirable for the decision maker, we do not consider supporting a decision maker to find a highly robust Pareto optimal solution in this thesis due to the existence issue.

Flimsly robust Pareto optimality aims at finding solutions which are Pareto optimal for one of the possible scenarios. It is originally proposed in [14] for multiobjective linear optimization problems and is generalized in [44]. Finding flimsly robust Pareto optimal solutions is equivalent to solving a deterministic multiobjective optimization problem under any possible scenario. Conventional interactive multiobjective optimization methods can support a decision maker to find a most preferred flimsly robust Pareto optimal solution.

3.2.3 Decision Support in Multiobjective Robust Optimization

As mentioned before, it is not always clear how to compute robust Pareto optimal solutions. The consideration of supporting a decision maker to find a most preferred robust solution is even rarer. Different possible scenarios are all considered at the same time in an interactive method in [61]. The decision maker is expected to select which possible scenarios to concentrate on in each iteration. The outcome of the solution in each scenario is presented to the decision maker. The main disadvantage in this approach is that the decision maker is forced to consider multiple scenarios simultaneously and this can be cognitively demanding.

In [39], the augmented weighted Chebyshev approach [75] is used to support the decision maker in finding a most preferred robust solution under cardinality uncertainty. The main disadvantage of this approach is that different solutions are computed based on different weights in the augmented weighted Chebyshev function. The decision maker is expected to judge if a solution is desirable or not. The rejected solutions are not recoverable. Also supporting the decision maker by effectively presenting the solutions for example with visualization is not considered in the paper.

In addition to proposing a robustness measure for decision uncertainty mentioned earlier, this thesis considers minmax robustness to optimize in the worst case and light robustness to seek for trade-off between robustness and quality. In addition, we also consider the solutions in the nominal scenario. We provide some solution methods for computing the robust Pareto optimal solutions for practical problems. We also analyze the relationships between different kinds of robust solutions aiming at gaining insights into utilizing different robustness concepts in decision making. More importantly, we consider how to support the decision maker to make informed decisions and balance between robustness and quality during the interactive solutions processes.

3.3 Fundamental Robustness Concepts for This Thesis

In this section, we summarize the robustness concepts considered in this thesis including set-based minmax robustness, point-based minmax robustness, and light robustness.

3.3.1 Set-based Minmax Robustness

Definition 5. *A solution $x^* \in X$ is set-based minmax robust Pareto optimal, if there does not exist $x \in X$ and $x \neq x^*$ such that $f_{\mathcal{U}}(x) \subseteq f_{\mathcal{U}}(x^*) - \mathbb{R}^k_{\geq}$, where $f_{\mathcal{U}}(x)$ is the outcome set of x as defined earlier.*

We denote the set of set-based minmax robust Pareto optimal solutions by X^{rpo}. For computing set-based minmax robust Pareto optimal solutions for (6), we compare the suprema of the outcome sets (of the feasible solutions) by solving the following problem:

$$\begin{aligned} &\text{minimize} && \sup_{\xi \in \mathcal{U}} f(x) = (f_1(x,\xi), ..., f_k(x,\xi))^T \\ &\text{subject to} && x \in X. \end{aligned} \tag{7}$$

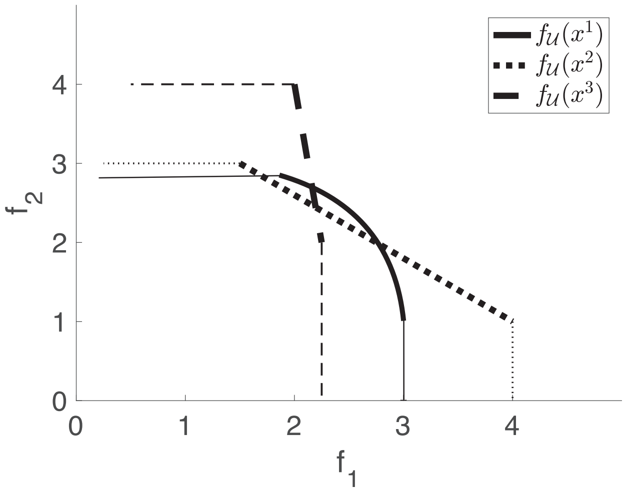


FIGURE 3 Example of set-based minmax robustness

Example 1. *Given a set of feasible solutions* $X = \{x^1, x^2, x^3\}$ *and an uncertainty set* $\mathcal{U}$*, we define the objective functions to be minimized* $f : X \times \mathcal{U} \mapsto \mathbb{R}^2$*. The outcome sets of the three feasible solutions are plotted in Figure 3 as bolded lines and a curve. The thin lines with same style as the corresponding outcome sets mark the borders of the outcome sets:* $f_{\mathcal{U}}(x^1) - \mathbb{R}^2_{\geq}$ *(solid curve and lines),* $f_{\mathcal{U}}(x^2) - \mathbb{R}^2_{\geq}$ *(dotted lines), and* $f_{\mathcal{U}}(x^3) - \mathbb{R}^2_{\geq}$ *(dashed lines). Solution* x^1 *is set-based minmax robust Pareto optimal because* $f_{\mathcal{U}}(x^1) - \mathbb{R}^2_{\geq}$ *does not contain* $f_{\mathcal{U}}(x^2)$ *nor* $f_{\mathcal{U}}(x^3)$*. Using the same way of observation, we can see that solutions* x^2 *and* x^3 *are also set-based minmax robust Pareto optimal.*

The definition of set-based minmax robustness is based on the set-based order (see e.g., [49]). In this thesis, we also utilize set-based orders in our comparison of different sets of solutions in [PIV] and comparing different outcome sets in [PIII]:

Definition 6. *Given two sets* $Y_1, Y_2 \in \mathbb{R}^k$ *, we have the following set-based orders:*

(i)

$$Y_1 \prec^{upp} Y_2 \text{ if for all } y \in Y_2 \text{ there exists } y' \in Y_1 \text{ with } y'_i \leq y_i \text{ for all } i = 1, ..., k$$

(ii)

$$Y_1 \prec^{low} Y_2 \text{ if for all } y \in Y_1 \text{ there exists } y' \in Y_2 \text{ with } y_i \leq y'_i \text{ for all } i = 1, ..., k.$$

With the help of a nonnegative ordering cone $\mathbb{R}^k_{\geq}$, we can write $Y_1 \prec^{upp} Y_2$ equivalently as

$$Y_1 \prec^{upp} Y_2 \text{ if } Y_2 \subseteq Y_1 + \mathbb{R}^k_{\geq}$$

which is known as the upper set less order. We can also write $Y_1 \prec^{low} Y_2$ equivalently as

$$Y_1 \prec^{low} Y_2 \text{ if } Y_1 \subseteq Y_2 - \mathbb{R}^k_{\geq}$$

which is known as the lower set less order.

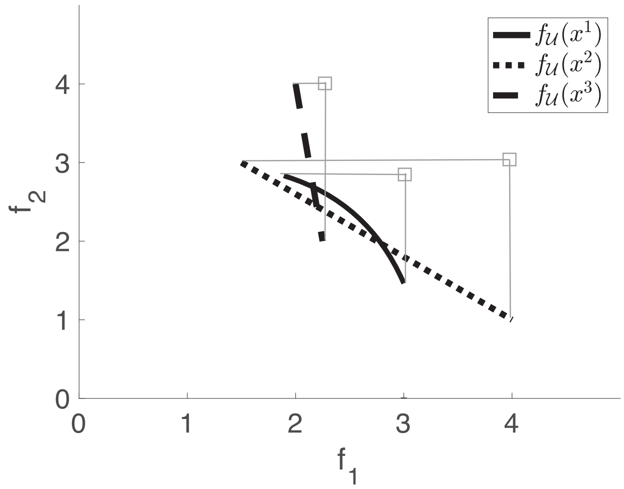


FIGURE 4 Example of point-based minmax robustness

3.3.2 Point-based Minmax Robustness

For point-based minmax robustness, we optimize the suprema of each objective function in the following form to transform (6) to a deterministic multiobjective optimization problem:

$$\begin{aligned} \text{minimize} \quad & f^{\text{wc}}(x) = (\sup_{\xi \in \mathcal{U}} f_1(x,\xi), ..., \sup_{\xi \in \mathcal{U}} f_k(x,\xi))^T \\ \text{subject to} \quad & x \in X. \end{aligned} \tag{8}$$

In this thesis, we assume the existence of the individual maxima of each objective function with a fixed $x \in X$. This means that we consider $\max_{\xi \in \mathcal{U}} f_i(x,\xi)$ for $i = 1, ..., k$. We name the objective functions as f^{wc} to indicate that this is what we mean by point-based worst case.

Definition 7. *A solution $x^* \in X$ is point-based minmax robust Pareto optimal, if there does not exist $x \in X$ and $x \neq x^*$ such that $\max_{\xi \in \mathcal{U}} f_i(x,\xi) \leq \max_{\xi \in \mathcal{U}} f_i(x^*,\xi)$ for all $i = 1, ..., k$ and for at least one objective j it holds that $\max_{\xi \in \mathcal{U}} f_j(x,\xi) < \max_{\xi \in \mathcal{U}} f_j(x^*,\xi)$.*

We denote the set of point-based minmax robust Pareto optimal solutions by X^{wc}. As analyzed in [15], point-based minmax robustness coincides with set-based minmax robustness and hull-based minmax robustness if (6) is an objective-wise uncertain problem.

Example 2. *Given the same set of feasible solutions, uncertainty set, and objective functions as in Example 1, we plot the outcome sets of the feasible solutions: $f_\mathcal{U}(x^1)$ (solid curve), $f_\mathcal{U}(x^2)$ (dotted line), and $f_\mathcal{U}(x^3)$ (dashed line) in Figure 4. The gray lines helps us to find the individual maxima of both objective functions in each outcome set. The points which are formed by the individual maxima are marked by the squares. Solution x^1 is point-based minmax robust Pareto optimal, because the corresponding square is not* *dominated by the squares corresponding to x^2 and x^3. Using the same method of observation, we can see that solution x^3 is also point-based minmax robust Pareto optimal. But, solution x^2 is not because its corresponding square is dominated by the square which corresponds to x^1.*

In both (7) and (8), we optimize in the worst case. The two concepts differ from each other in the representations of the worst case. In order to identify the worst case objective function values of a fixed feasible solution x, we need to solve the following multiobjective problem with the objectives to be maximized:

$$\begin{array}{ll} \text{maximize} & (f_1(x,\xi), ..., f_k(x,\xi))^T \\ \text{subject to} & \xi \in \mathcal{U}. \end{array} \tag{9}$$

Since (9) is a multiobjective optimization problem, there usually does not exist a single worst case scenario. Instead, there is a set of worst case scenarios which consist of the Pareto optimal solutions of (9). As analyzed in [26], if (6) is an objective-wise uncertain problem, there is a single worst case scenario.

For the decision maker to understand the ranges of possible outcomes in the worst case, we can utilize the worst case ideal objective vector and the worst case nadir objective vector. The worst case ideal objective vector is formed by the optimal value of each objective function in the worst case and the worst case nadir objective vector can be approximated by the payoff table.

3.3.3 Nominal Solutions and Light Robustness

As mentioned before, the objective function values of the minmax robust Pareto optimal solutions can be bad under other scenarios. Light robustness is a concept developed based on the idea of controlling the trade-off between the robustness and the degradation in objective function values.

In light robustness, we assume the existence of a nominal scenario $\hat{\xi}$ which describes the most typical behavior of the uncertain parameters. For finding the Pareto optimal solutions in the nominal scenario, we solve the following problem:

$$\begin{array}{ll} \text{minimize} & f^{\text{nom}}(x) = (f_1(x,\hat{\xi}), ..., f_k(x,\hat{\xi}))^T \\ \text{subject to} & x \in X. \end{array} \tag{10}$$

In the formulation, we named the objective functions f^{nom} to help us to refer to the problem under nominal scenario more conveniently.

Definition 8. *We call the solutions of* (10) *the nominal Pareto optimal solutions.*

The set of nominal Pareto optimal solutions is denoted by X^{nom}. For a solution $x \in X$, we define its nominal quality and its nominal outcome as $f^{\text{nom}}(x)$. For finding lightly robust Pareto optimal solutions, we optimize in the worst case with an acceptable degradation on the objective function values when compared to a nominal Pareto optimal solution in the nominal scenario. We couple light robustness with point-based minmax robustness in this thesis. For finding lightly

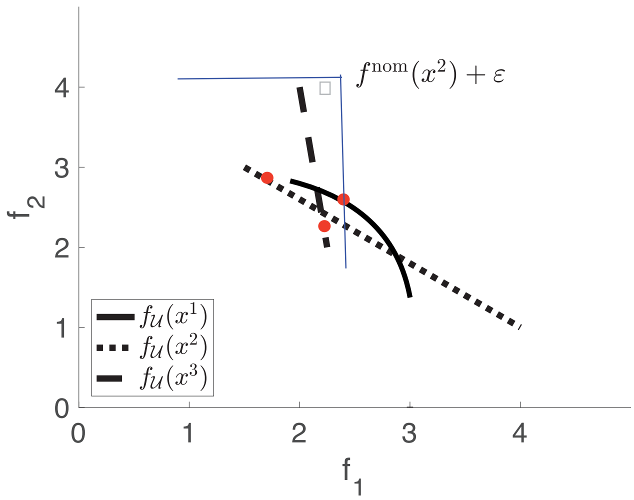


FIGURE 5 Example of light robustness

robust Pareto optimal solutions, we solve the following problem (under the assumption that individual maximum of each objective exists for a fixed $x \in X$):

$$\begin{array}{ll} \text{minimize} & (\max_{\xi \in \mathcal{U}} f_1(x,\xi), ..., \max_{\xi \in \mathcal{U}} f_k(x,\xi))^T \\ \text{subject to} & x \in X \\ & f_i(x,\hat{\xi}) \leq f_i(\hat{x},\hat{\xi}) + \varepsilon_i \text{ for all } i = 1, ..., k, \end{array} \tag{11}$$

where $\hat{x}$ is a nominal Pareto optimal solution and the k-dimensional vector ε represents the tolerable degradations on the objective function values in the nominal scenario. We call the constraints $f_i(x,\hat{\xi}) \leq f_i(\hat{x},\hat{\xi}) + \varepsilon$ for all $i = 1, ..., k$ the lightly robust constraints.

Definition 9. *The solutions of* (11) *are lightly robust Pareto optimal solutions.*

We denote the set of lightly robust Pareto optimal solutions by X^{light}.

Example 3. *Given the same set of feasible solutions, uncertainty set, and objective functions as in Example 1, we plot the outcome sets of the three solutions: $f_{\mathcal{U}}(x^1)$ (solid curve), $f_{\mathcal{U}}(x^2)$ (dotted line), and $f_{\mathcal{U}}(x^3)$ (dashed line). The nominal outcomes of the three solutions are marked by the circles. The nominal Pareto optimal solution identified for finding lightly robust Pareto optimal solutions is $f^{\text{nom}}(x^2)$. The blue lines mark the borders of the lightly robust constraints with the corner as $f^{\text{nom}}(x^2) + \varepsilon$. The point-based worst case outcome of x^3 (marked by the small gray square) is inside the blue lines, thus feasible for problem* (11)*. Since it is the only feasible solution, it is the lightly robust Pareto optimal solution.*

4 SOLVING MULTIOBJECTIVE OPTIMIZATION PROBLEMS UNDER DECISION UNCERTAINTY WITH AN INTERACTIVE METHOD

4.1 Motivation

We start the research by studying supporting a decision maker under decision uncertainty. On the one hand, it is merely because decision uncertainty is one type of uncertainty considered in this thesis. On the other hand, it can be easier for the decision maker to access information related to decision uncertainty. As mentioned before, decision uncertainty reflects the case that the implementation of solutions involves some errors, so it usually comes from, e.g., machines used in production. The decision maker can get such kind of information from, e.g., documentation of production machines.

As mentioned in Chapter 3, a common way of handling decision uncertainty is to study the neighborhood of nominal solutions and use a robustness measure to quantify the effects of uncertainty on the objective function values. Alternatively, some of the concepts on robust Pareto optimality, e.g., minmax robust Pareto optimality, can also be applied. Based on the source of decision uncertainty described earlier, it is more desirable for the decision maker to consider solutions based on the nominal quality and deviation from the nominal quality than consider e.g., the behaviors of solutions in the worst case. With this line of consideration, the decision maker can be supported to find solutions which are with good objective function values and at the same time with satisfactory deviation.

Based on the consideration described above, Paper [PI] follows the approach of using a robustness measure to quantify the effects of uncertainty in the neighborhood of the nominal solutions. The goal is not to develop an entirely new interactive method. Instead, the aim is to augment existing methods to be able to support a decision maker to find a final solution for (5). We augment the synchronous NIMBUS method [60] as an example. The same principle can be applied in, for example, the reference point method [81].

For supporting a decision maker, we summarize desired properties of a robustness measure as:

1. The numerical value should reflect the effects of decision uncertainty on the objective function values. This numerical value should be informative to the decision maker to indicate how "robust" a solution is.
2. Having seen the numerical value, the decision maker should be able to formulate and specify her or his preferences conveniently.

We studied the existing robustness measures in the literature with respect to the desired properties and identified three measures D-G [22], G-C [30], and WCSR [54]. The three measures are the closest to the desired properties. The measures D-G and G-C study how the objective function values change in the neighborhood of a nominal solution. The smaller the change is, the more "robust" the nominal solution is. The measure WCSR quantifies the maximum perturbations the nominal solution can have such that the objective function values are within a pre-specified tolerance.

In the three measures, the decision maker can only rely on one's intuition on the numerical values which somehow represent the robustness of a solution. With the numerical values, the decision maker cannot get any information of how the involved uncertainty affects the objective function values. In addition, the numerical measures do not have any unit. For example, it is hard for the decision maker to understand how much better value 0.1 is compared to 0.2. All these issues bring challenges to the decision maker in formulating her or his preferences for a more desirable solution.

4.2 Proposed Robustness Measure

Inspired by the literature, we propose a new robustness measure. Our robustness measure provides concrete information in terms of how decision uncertainty affects the objective function values of a nominal solution. The new robustness measure is based on the ranges of the objective function values in the neighborhood. The ranges can be represented by the worst objective function values and the best objective function values in the neighborhood of a nominal solution. For an objective function f_i, its range in the neighborhood of the nominal solution x^b is formulated as follows:

$$r_i(x^b) = \max_{\Delta x \in \mathcal{U}} f_i(x^b + \Delta x) - \min_{\Delta x \in \mathcal{U}} f_i(x^b + \Delta x),$$

where Δx represents the possible perturbations of the nominal solution x^b and $\mathcal{U}$ represent the neighborhood. In this way, we utilize the unit of the objective function values in quantifying robustness. Naturally, $r_i(x^b)$ is defined for each objective function. We can use the formulation directly as k additional objectives to introduce trade-offs between nominal quality and robustness. However, using the formulation directly means that we expect the decision maker to consider $2 \times k$ objective functions. In order to minimize the additional cognitive load introduced when considering robustness, we formulate a single robustness measure based on $r_i(x^b)$ and use it as an additional objective:

$$R(x^b) = \max_i \left[\frac{r_i(x^b)}{z_i^{\text{nad}} - z_i^{\text{uto}}} \right], i = 1, ..., k.$$

With $R(x^b)$, we provide information of the normalized maximum change of the objective function values for the decision maker. This value is the maximum percentage of changes in the objective function values in the neighborhood when compared to the utopian and nadir values of the objectives. Together with z^{nad} and z^{uto}, the decision maker can understand the magnitude of changes in the objective function values. In addition, we provide the nominal objective function values. Combining the measured robustness and the nominal objective function values of a solution, the decision maker can concretely understand how uncertainty affects the objective function values.

4.3 Proposed Interactive method

As mentioned earlier, we incorporate $R(x^b)$ to the problem formulations as an additional objective. We build an interactive method for supporting a decision maker to find a most preferred solution concerning both the nominal objective function values and the change of objective function values due to decision uncertainty. The idea is that by having robustness as an additional objective, the decision maker can balance between robustness and other objective function values. As mentioned before, we augment the synchronous NIMBUS method to build the proposed approach. The synchronous NIMBUS method is briefly described in Chapter 2. We will discuss the changes made next.

In interactive solution processes of problems, we calculate a current solution in each iteration based on the decision maker's preferences. Then, we present it to the decision maker. When we present the current solution to the decision maker, we visualize the following information:

1. The nominal ideal and nadir values of each objective function.
2. The nominal objective function values of the current solution.
3. The value of $R(x^b)$ in the corresponding objective, which we call the active objective.
4. The value of $r_i(x^b)$ of each objective.

We augment the IND-NIMBUS [64] visualization as shown with an example in Figure 6. In the figure, there are two objective functions to be minimized. The white bars represent the ranges, that is, the nominal ideal (z_i^*) and nadir (z_i^{nad}) objective function values while the filled blue bars represent the nominal objective

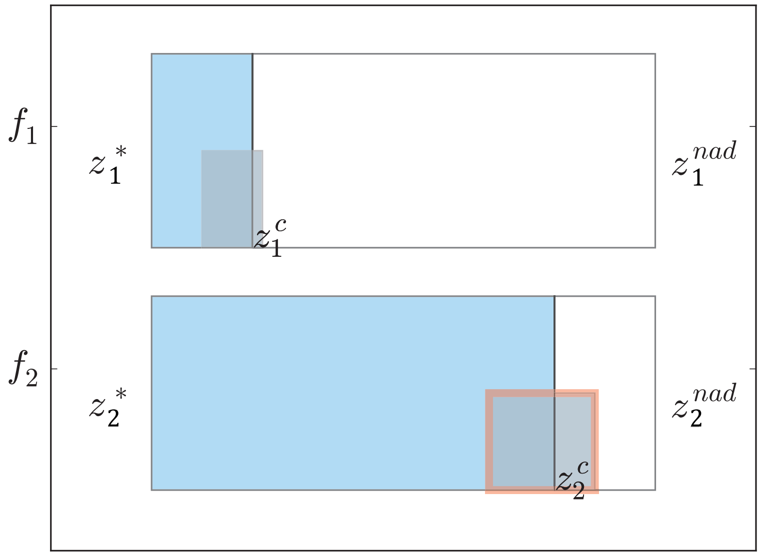


FIGURE 6 Visualization with robustness information

function values. The nominal objective function values are marked at the ending of the blue bars by z_i^c. The values of $r_i(x^b)$ are illustrated with the gray bars (which we call shadows) representing the upper and lower bounds of the changes of the objective function values due to the perturbations of x^b in the neighborhood $\mathcal{U}$. The shadow of the active objective is highlighted with a frame in red and the value of robustness measure is illustrated on the highlighted shadow.

When a decision maker is shown a visualized solution as in Figure 6, (s)he can consider what kind of changes are needed to obtain a more desirable solution. The decision maker is expected to classify the original objective functions as in the NIMBUS method. As for the robustness measure, we provide the opportunity for the decision maker to choose a more comfortable way of specifying her or his preferences either by classifying it or by providing a desired value for it. When the decision maker has a preference on what kind of changes (e.g., increase or decrease) the robustness measure should have, (s)he can classify the robustness measure. In this case, we solve (4) to find a new solution. When the decision maker has a concrete desired value for the robustness measure, (s)he can choose to specify an aspiration level. In this case, we convert the aspiration level to a classification depending on the comparison of the aspiration level and the current value of the robustness measure. The details on the conversion are described in Paper [PI]. After the conversion, we combine the classification of the original objectives and the converted class of the robustness measure and solve (4) to find a new solution for the decision maker. The solution process continues until the decision maker finds a most preferred solution. The general steps of the interactive method are presented in Figure 7. In the figure, the changes made to the original NIMBUS method discussed above are highlighted with red dashed lines and decision maker is abbreviated as DM.

In order to demonstrate the ability of the proposed method in supporting a decision maker to find a most preferred solution regarding the nominal qual-

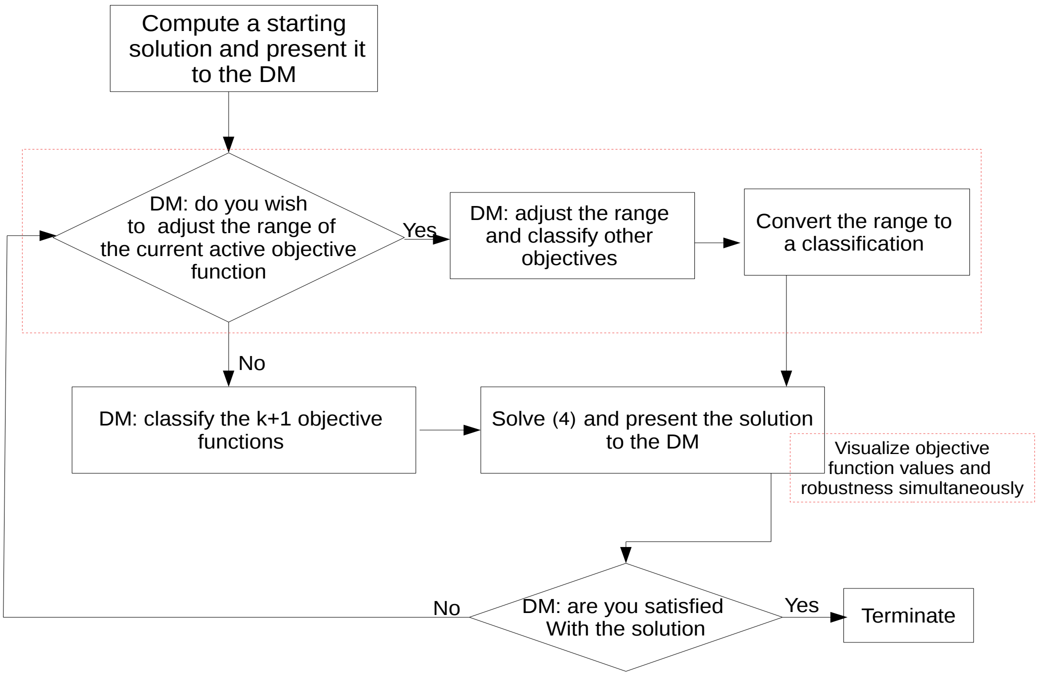


FIGURE 7 Flowchart of the interactive approach

ity and robustness, we solved two different types of example problems in Paper [PI]. We first solved a river pollution management problem which is originally presented in [62]. In the problem, we assumed that the decision variables which represent the amount of biochemical matters to be removed from the water in two water treatment plants cannot be realized exactly. Then we formulated a multiobjective version of a procurement contract selection problem with pricing optimization for process networks and solved the problem for an example process network. The decision variables which represent the amount of raw material consumed in each processing plant were considered to be uncertain.

The solution processes of both problems involved real decision makers. Both of the solution processes demonstrated the advantages of the proposed approach. The decision makers were able to understand the effects of decision uncertainty based on the value of the robustness measure and the related information on the changes of each objective function. The decision makers were also able to consider the nominal objective function values and the robustness of the solutions at the same time. With the interactive solution processes, the decision makers eventually learned how the objective function values and the values of the robustness measure affected each other. Thus, this interactive method fulfilled our goal to better support a decision maker to make informed decisions when decision uncertainty is involved in a multiobjective optimization problem.

It is important to note that it is hard to verify if the solution we find in the proposed interactive method is Pareto optimal or not to the original problem. According to the discussion in [78], this depends on how the robustness measure conflicts with the original objective functions. As mentioned in [12], usually, some sacrifices on the nominal objective function values are required to gain robustness. Thus, losing the Pareto optimality of a solution could be the price that the decision maker is willing to pay in order to gain robustness.

5 FINDING ROBUST SOLUTIONS UNDER PARAMETER UNCERTAINTY WITH AN INTERACTIVE METHOD

5.1 Motivation

Paper [PI] addresses the issue of supporting a decision maker to find a most preferred solution under decision uncertainty. The other type of uncertainty considered in this thesis is parameter uncertainty in the objective functions. This kind of uncertainty is usually due to imprecise data or uncertain future development. As mentioned before, it can be more challenging for the decision maker to access information on the uncertainty set. As an expert in the problem domain, the decision maker can utilize her or his experiences or domain knowledge in providing information on the uncertainty set. In this situation, the decision maker can seek for solutions which help her or him to be well prepared for the worst possible cases. Based on the domain knowledge, the decision maker can identify a nominal scenario. Since the nominal scenario describes the most typical behaviors of the uncertain parameters, the decision maker can be more focused on it while making decisions.

In order to support the decision maker in the consideration of the two aspects, i.e., preparing for the worst case and focusing on the nominal scenario while making decisions, we develop the MuRO-NIMBUS method in Paper [PII] utilizing set-based minmax robustness. The definition of set-based minmax robust Pareto optimality was published in [26]. The application in a wood cutting problem [45] has shown potential of this concept to solve practical problems. But, applications of this concept to supporting decision making have not yet been published. The MuRO-NIMBUS method brings this into the context of decision making. The MuRO-NIMBUS consists of the pre-decision making and the decision making stages. The reason is that considering both the worst case and the nominal scenario at the same time requires solving two types of problems simultaneously. As a result, the decision maker also has to deal with the two types of solutions at the same time.

Any set-based minmax robust Pareto optimal solution can serve the purpose of helping the decision maker to be well prepared for the worst case. We do not need to involve the decision maker in the computation of the minmax robust Pareto optimal solutions (by e.g., asking for preferences). In the pre-decision making stage, we compute a representative set of set-based minmax robust Pareto optimal solutions without the involvement of the decision maker. Only in the nominal scenario, which the decision maker focuses on when making decisions, the preferences of the decision maker are needed. In the decision making stage, we support the decision maker to find a set-based minmax robust Pareto optimal solution which corresponds to the most preferred objective function values in the nominal scenario.

5.2 Robust Version of Achievement Scalarizing Function

For the pre-decision making stage, we propose a robust achievement scalarizing function (ASF) approach. Based on (7) and (2), the robust version of the subproblem based on the achievement scalarizing function is as follows:

$$\begin{aligned} &\text{minimize} && \sup_{\xi \in \mathcal{U}} \max_i [w_i(f_i(x,\xi) - \bar{z}_i)] + \rho \sum_{i=1}^{k} w_i(f_i(x,\xi) - \bar{z}_i) \\ &\text{subject to} && x \in X. \end{aligned} \tag{12}$$

Just like (2), the robust version also involves a reference point $\bar{z}$ and a projection weight w. The following theorem states the necessary and sufficient conditions for an optimal solution of (12) to be set-based minmax robust Pareto optimal:

Theorem 10. *Given an uncertain multiobjective optimization problem* (6), $x^* \in X$ *is a set-based minmax robust Pareto optimal solution, if and only if* x^* *is an optimal solution to* (12) *for some* $\bar{z}$ *and* w*, and* $\max_{\xi \in \mathcal{U}} f_i(x,\xi)$ *exists for all* $x \in X$ *and for all* $i = 1, ..., k$.

The proof of the sufficient condition is given in Paper [PII]. Since the objective function in (12) is a strongly increasing function, it is also a strictly increasing function. Theorem 4.4 in [15] states that a solution is set-based minmax robust Pareto optimal only if it is an optimal solution of a strictly increasing scalarizing function.

Figure 8 illustrates an example of solving subproblem (12) to find a set-based minmax robust Pareto optimal solution. Given the same problem as in Example 1, we specify a reference point $\bar{z}$ marked as the black circle in the figure. The arrowed line represents the projection direction specified by w. The two solid lines which intersect at the reference point represent the cone $\bar{z} - \mathbb{R}^2_\rho$. Since x^1, x^2 and x^3 are all set-based minmax robust Pareto optimal, the solution whose outcome set intersects first with the cone $\bar{z} - \mathbb{R}^2_\rho$ along the projection direction is the solution which satisfies the reference point best. In this example, it is x^2.

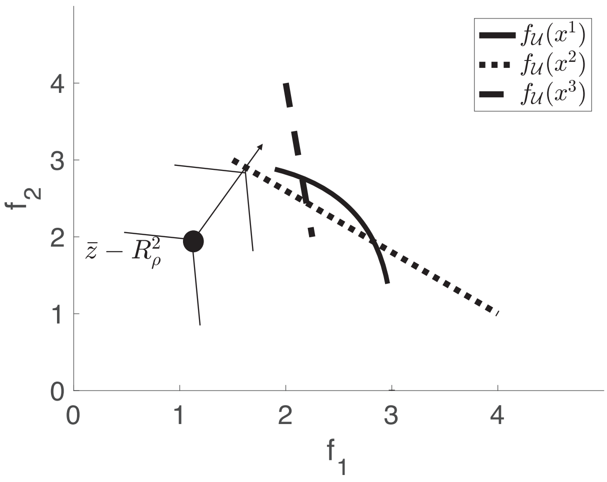


FIGURE 8 Example of robust ASF

Based on (12), we can compute a set of set-based minmax robust Pareto optimal solutions by varying the reference point $\bar{z}$ or the projection weights w. As we use a pre-computed set of solutions for decision making, we should obtain a good representative set of solutions. For example, in [19], reference vectors are used in an evolutionary multiobjective optimization method to associate objective vectors in different parts of the objective space. Reference vectors are formed by generating equally distributed points on an unit hyper plane in the objective space. In the robust ASF approach, we utilize this idea to vary the reference point and projection weights. We generate a set of equally distributed points as reference points, and use the vector from the utopian objective vector to the reference point as the projection weights.

The robust ASF approach takes (6) as input and calculate a set of robust Pareto optimal solutions X^{rpo} as output. The steps of the robust ASF approach is as follows:

Step 1. Set $X^{\text{rpo}} = \emptyset$ and generate a set of reference points $\bar{\mathfrak{Z}}$.
Step 2. If $\bar{\mathfrak{Z}} = \emptyset$, stop.
Step 3. Choose $\bar{z} \in \bar{\mathfrak{Z}}$, and set $\bar{\mathfrak{Z}} = \bar{\mathfrak{Z}} \setminus \{\bar{z}\}$.
Step 4. Find an optimal solution x^* to (12) using $\bar{z}$ as the reference point and set w accordingly, e.g., $w_i = \frac{1}{z_i^{\text{uto}} - \bar{z}_i}$, where z^{uto} is the utopian objective vector. Set $X^{\text{rpo}} = X^{\text{rpo}} \cup \{x^*\}$.
Step 5. Go to step 2.

5.3 The MuRO-NIMBUS Method

The MuRO-NIMBUS method includes the robust ASF approach presented above in the pre-decision making stage and a decision making stage based on the synchronous NIMBUS method [60]. The steps of the MuRO-NIMBUS method are as follows:

1. **Pre-decision making.**
   (a) Calculate the set X^{rpo} with the robust ASF approach. Calculate also the nominal ideal and nadir objective vectors.
2. **Decision making**
   (a) Classify all the objectives into the class $I^{<}$ of the NIMBUS classification and solve (4) (by including only the first constraint) to find an initial set-based minmax robust Pareto optimal solution x^c, whose nominal outcome is non-dominated in the set X^{rpo}. Note that the feasible set in (4) here is X^{rpo}.
   (b) Present the nominal ideal and nadir objective vectors to the decision maker.
   (c) Present the outcomes in the nominal and worst cases corresponding to x^c to the decision maker. If the decision maker is satisfied, x^c is the final solution. Otherwise, continue.
   (d) Ask the decision maker to classify the objectives at the current solution based on the outcome in the nominal case. Then solve (4) to find a new solution and set it as x^c and go to step 2(c).

As mentioned before, the pre-decision making stage does not require the presence of the decision maker. In the decision making stage, we start from finding and presenting to the decision maker a set-based minmax robust Pareto optimal solution whose outcome is non-dominated in the nominal case together with the nominal ideal and nadir objective vectors. Note that in Step 2(c), we also present the worst case outcome. This is only an additional piece of information for the decision maker to make informed decisions. The focus, i.e., specifying the preferences, is on the nominal outcome. If (s)he feels that it is not necessary or the cognitive load is getting heavy, the information about the worst case outcome can be ignored.

In order to help the decision maker to understand the nominal outcome and the worst case outcome of the presented solution, we augment the value path visualization [57]. Figure 9 is an example of the visualization. In the figure, vertical bars represent the ranges, i.e., the nominal ideal and nadir values of each objective function. The value path represents the nominal outcome. The objective function values in the worst case are marked by the gray vertical bars inside each objective. In case there is only one worst case scenario, the worst case objective function values can be marked with triangles. With this visualization, the decision maker observes not only the outcomes of the solution, but also the differences between the nominal and worst case outcomes.

After having seen a solution, if the decision maker is not satisfied, (s)he is expected to classify the objectives into the NIMBUS classification by considering the nominal objective function values. The information on the worst case outcome is for the decision maker to be aware of what to expect in the worst case. Based on the classification, we find a new solution from the set X^{rpo} and present

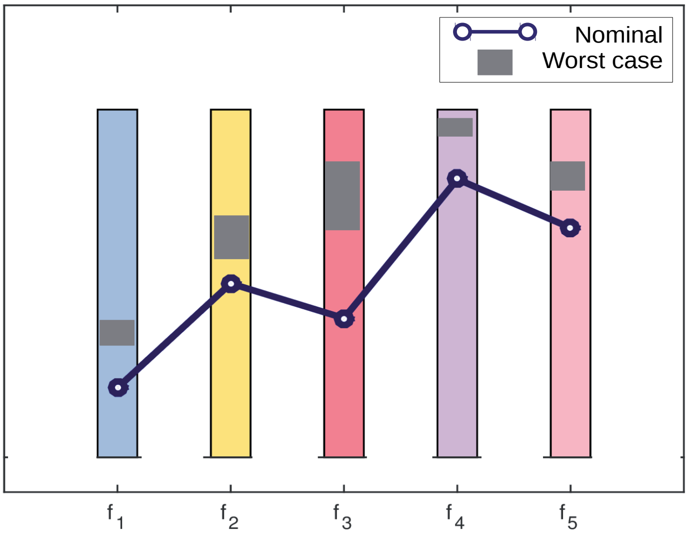


FIGURE 9 Visualization of a set-based minmax Pareto optimal solution

it to the decision maker. If the decision maker is satisfied, (s)he takes the current solution as the final solution, and we terminate the solution process.

We demonstrated in [PII] the solution process of a multiobjective ship design problem where two parameters involved interval uncertainty by applying the MuRO-NIMBUS method. The deterministic formulation of the problem has been presented in [84]. We first computed a set of set-based minmax robust Pareto optimal solutions and then supported a decision maker to find a most preferred one in the decision making stage. During the decision making stage, the decision maker was able to consider the nominal outcomes and observe the worst case outcomes. She was able to direct the solution process towards the most preferred solution by classifying the objectives based on the nominal outcomes. Even though she was not able to interfere the outcomes in the worst case, she was able to utilize them as background information to make informed decisions. So the final solution found for her was most preferred in terms of the nominal objective function values. At the same time, it is still valid even if the worst case happens.

5.4 Discussion Related to MuRO-NIMBUS

In MuRO-NIMBUS, we utilized subproblem (12) to obtain the set X^{rpo} in the predecision making stage. The utilization of subproblem (12) is not limited here. It can be used in the reference point approach when a decision maker wants to concentrate on the worst case and find a most preferred solution based on the worst case outcomes. In this setting, the decision maker can provide a reference point which consists of desired values of objective functions in the worst case. With the reference point, we can solve (12) to find a set-based minmax robust Pareto optimal solution which satisfies the reference point as well as possible and present it to the decision maker. The decision maker can iteratively specify reference points until (s)he finds a set-based minmax robust Pareto optimal solution with a most preferred worst case outcome.

There are two issues in the pre-decision making stage of MuRO-NIMBUS. One is the number of the pre-computed set-based minmax robust Pareto optimal solutions. The other is related to solving (12). For the number of pre-computed solutions, it depends on the problem, for example how the set-based minmax Pareto optimal solutions vary in the nominal case. For the multiobjective ship design problem, 150 solutions were regarded as sufficient. With lower number of solutions, it is frequently the case that we result in a same solution if the preferences between two iterations are not significantly different. With 150 solutions, the solutions are diverse enough for the decision making stage to respond to the decision maker's preferences. As far as solving (12) is concerned, it is a very challenging task. We need to solve a maximization problem for each feasible solution evaluated. In the robust ASF approach used to solve the multiobjective ship design problem, we took samples in the uncertainty set and used them to replace the uncertainty set.

Using samples in solving (12) might not give us accurate results because it is not guaranteed the samples include the ξ which gives the maximum of the scalarized objective function for a fixed $x \in X$. In addition, we also need to solve multiple instances of (12) to obtain X^{rpo}. As a further development of the pre-decision making stage, we propose the simple indicator-based evolutionary algorithm for set-based minmax robustness (SIBEA-R) to approximate a set of set-based minmax robust Pareto optimal solutions in Paper [PIII].

5.5 The SIBEA-R Method

So far, we have discussed interactive methods which utilize scalarized subproblems to compute Pareto optimal solutions for multiobjective optimization problems. As mentioned before, we can also calculate an approximated set of solutions with evolutionary multiobjective optimization methods. The output of an evolutionary multiobjective optimization method is a set of non-dominated solutions. We propose the SIBEA-R method in [PIII] for improving the pre-decision making stage in MuRO-NIMBUS.

The SIBEA-R method compares the worst case outcomes using set-based order and is based on the SIBEA method [87]. In the literature, evolutionary multiobjective optimization methods are proposed based on different ideas. The decomposition based methods (e.g., MOEA/D [86]) require associating objective vectors to different weights based on e.g., distance to the weight vector. In our

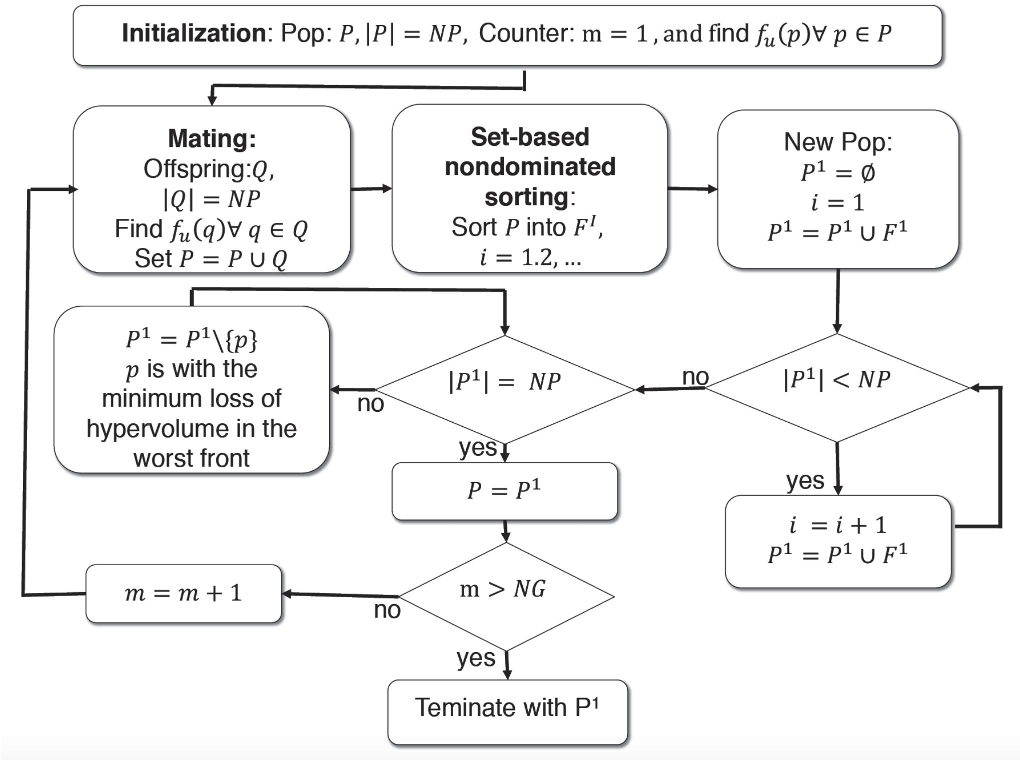


FIGURE 10 The steps of SIBEA-R

case, we cannot easily associate a set of worst case outcomes to some weights. The methods based on non-dominated sorting and crowding distance (e.g., NSGA-II [24]) compares the solutions in the same front based on some distance. In our case, we would have needed to comparing the distance between different sets of worst case outcomes. In indicator-based methods, outcomes are compared based on some indicator function. The relationships between comparing individual outcomes and different sets of outcomes using an indicator function are well established in the literature, e.g., in [5, 89]. After exploring the possibilities, we extend SIBEA which allows us to compare different sets of worst case outcomes using the hypervolume indicator.

The SIBEA-R method takes the population size (NP) and the number of generations (NG) as the input and produces a set of non-dominated set-based min-max robust solutions A as the output. The basic steps of SIBEA-R are illustrated in Figure 10. We extend the SIBEA method by two components: finding the representative worst case outcomes and the set-based non-dominated sorting.

The worst case outcomes of each solution are considered as a set. We compare the sets of worst case outcomes corresponding to different solutions with set-based dominance. As mentioned before, for a fixed solution, we need to maximize the objectives simultaneously with respect to the uncertainty parameters (i.e., solve (9)) to find the worst case outcomes. We can employ an evolutionary multiobjective optimization method to find the approximated worst case outcomes. But, doing so requires computational resources. In addition, comparing the worst case outcome sets using set-based dominance requires us to consider each elements in the sets. This means that we should limit the number of worst case outcomes in the set-based non-dominated sorting so that we can rank the solutions in reasonable time.

So, in SIBEA-R, we propose to solve a small number (M) of scalarized function, e.g., (3) to find a representative set of worst case outcomes and use them in the set-based non-dominated sorting. The value of M is a parameter in SIBEA-R. The bigger the value is the more worst case outcomes can be found. As a result, the comparison of the sets can be more accurate. However, more computational resources are needed. The number of function evaluations of SIBEA-R depends on the single-objective optimization solver used to solve the scalarized subproblems. In case that there is a set of discrete scenarios in the uncertainty set, the number of function evaluations is $k \times NP \times NG \times$ number of scenarios.

The set-based non-dominated sorting is an extension of the non-dominated sorting presented in [24]. Instead of using normal Pareto dominance in the sorting, we use lower set less order. The steps of set-based non-dominated sorting are summarized as follows :

Compare individuals:
1. For each solution $p \in P$, find the representative set of worst case outcomes $f^{\mathrm{wc}}(p)$ and compare with other solutions $q \in P \setminus \{p\} : f^{wc}(q)$.
2. If $f^{\mathrm{wc}}(q) \prec^{low} f^{\mathrm{wc}}(p)$, increase the domination count of p, n_P by 1.
3. If $f^{\mathrm{wc}}(p) \prec^{low} f^{\mathrm{wc}}(q)$, put q to S_p, the set of solutions dominated by p.
Sort individuals into fronts:
1. If $n_p = 0$, then p is in the first front F^1. Set $i = 1$.
2. for each $p \in F^i$, set $n_q = n_q - 1$ for each $q \in S_p$. If $n_q = 0$, then add q to F^{i+1}.
3. Set $i = i + 1$ and repeat step 2 for $p \in F^{i+1}$.

It is very challenging to verify if SIBEA-R can find a reasonable set of approximated set-based minmax robust Pareto optimal solutions. In Paper [PIII], we calculated a set of set-based minmax robust Pareto optimal solutions with the robust version of the weighted-sum approach [26] for a linear problem. We compared these solutions with the final generations in 20 runs of the SIBEA-R method with different initial populations. Even though evolutionary multiobjective optimization methods are not meant for linear problems, we can use the solutions computed by the weighted-sum approach as references. The results showed that the SIBEA-R method was able to find those solutions. In addition, SIBEA-R also found some solutions which the robust weighted-sum approach was not able to find. These solutions can be verified by the definition to be non-dominated set-based minmax robust solutions.

5.6 Using SIBEA-R in Pre-decision Making in MuRO-NIMBUS

As SIBEA-R is developed for more efficient calculation of set-based minmax robust solutions in the pre-decision making stage of the MuRO-NIMBUS method, we also demonstrate in [PIII] how it can be used. We modified a common test problem ZDT2 (see, e.g., [21]) with two assumed uncertain parameters from a polyhedral uncertainty set. We first approximated a set of set-based minmax robust Pareto optimal solutions with the SIBEA-R method, and then we use the reference point approach to help a decision maker to find a most preferred solution based on the outcomes in the nominal case. We did not apply the classification based interactive method as in MuRO-NIMBUS in this example since the ZDT2-based problem we used was a bi-objective optimization problem. The decision maker could operate without much difficulty by considering the objective function values.

The development SIBEA-R not only enriched the solution methods for computing set-based minmax robust Pareto optimal solutions. For MuRO-NIMBUS, SIBEA-R improves the pre-decision making process by more accurately approximate the set of set-based minmax robust Pareto optimal solutions than the initially used approach based on sampling. Overall, SIBEA-R improves the integrity of MuRO-NIMBUS. MuRO-NIMBUS is not capable of supporting the decision maker to consider the trade-offs between robustness and nominal quality. In fact, the analysis on the trade-offs between robustness and nominal quality in multiobjective setting is not a commonly studied topic. In [PIV], we study the trade-offs based on the analysis of different solutions in the nominal case and the worst case.

6 BALANCING BETWEEN ROBUSTNESS AND NOMINAL QUALITY: ANALYZING SOLUTION SETS

6.1 Motivation

As mentioned before, minmax robustness is rather conservative. The conservativeness means that the minmax robustness helps the decision maker to prepare for the worst case, but the objective function values can be bad compared to other solutions in other scenarios. This means that the minmax robust Pareto optimal solution the MuRO-NIMBUS method supports the decision maker to find can have a bad nominal outcome.

As discussed before, the MuRO-NIMBUS method does not have the capability of allowing the decision maker to consider the trade-off between robustness and nominal quality. The trade-off between robustness and nominal quality can mean giving up some nominal quality to gain some robustness. Before introducing this kind of feature into an interactive method, we first need to investigate which robustness concepts can be utilized. We also need to understand the relationships among these robustness concepts.

In single-objective optimization, many concepts have been proposed (e.g., in [7, 12, 28, 35, 71]) to seek for less conservative solutions than minmax robust optimal solutions. For single-objective optimization problems, the uncertainty considered is usually reflected in the parameters in the constraints. The analysis on the trade-offs has also been focused on the “price” to be paid for feasibility.

As said, we consider multiobjective optimization problems with uncertain parameters in the objective functions. So, we consider the “price” to be paid for finding solutions that are good when different scenarios are considered simultaneously. In multiobjective optimization problems with parameter uncertainty, some less conservative concepts than minmax robustness have been developed including regret robustness (e.g., [83]), the extension of considering cardinality-uncertainty (e.g., [67]) and light robustness [44].

Multiobjective regret robustness is to find a minmax robust Pareto optimal solution whose nominal outcome is the nearest to that of a nominal Pareto optimal solution. If we fix the nominal Pareto optimal solution, the decision maker's preferences on the nominal outcome can be incorporated. But, the solution found is still a minmax robust solution without the trade-off between robustness and nominal quality. We have already considered supporting the decision maker to find a most preferred minmax robust Pareto optimal solution based on her or his preferences in the nominal scenario in MuRO-NIMBUS.

The consideration of cardinality-uncertainty is based on the idea that not all uncertain parameters attain the worst case at the same time. The decision maker can specify her or his preferences as how many parameters can involve uncertainty at the same time. Treatment of this information for trade-off between robustness and nominal quality is not automatically included in this kind of approach.

Light robustness proposes an idea of controlling the tolerable degradation in the nominal quality to gain robustness. The tolerable degradation is represented by the values ε. The robust solutions are found by optimizing in the worst case and utilizing ε to constrain the degradation of their nominal quality. As introduced in Chapter 3, the formulation of light robustness allow us to first fix the nominal outcome with respect to which the tolerable degradation is considered. This allows us to incorporate the decision maker's preferences on the nominal outcome. In addition, the decision maker can specify the value of ε to specify how much nominal quality (s)he can sacrifice to gain robustness.

Based on the discussion above, we choose light robustness for introducing trade-off between robustness and nominal quality. The intuition behind the light robustness concept is that the more nominal quality is sacrificed, the more robustness can be gained. In order to verify our intuition and justify that lightly robust solutions are good trade-offs between robustness and nominal quality, we analyze the nominal, lightly robust, minmax robust Pareto optimal solutions in Paper [PIV]. The nominal Pareto optimal solutions have the best nominal quality. The minmax robust Pareto optimal solutions are considered to be the most robust solutions.

In [44], the existence of ε such that lightly robust Pareto optimal solutions are set-based minmax robust Pareto optimal has been discussed. However, a systematical analysis of the relationships between the three sets of solutions when considering robustness and nominal quality is still missing in the literature. The set of nominal Pareto optimal solutions contains objective vectors. If we use set-based worst case in light robustness and minmax robustness, we need to deal with the comparison of set of outcome sets and a set of objective vectors. So, in our analysis, we consider point-based worst case. This allows us to compare sets consists of objective vectors.

In Paper [PIV], we analyze the three sets of solutions using set-based dominance from set-valued optimization. With the analysis, we first verify that lightly robust Pareto optimal solutions are good trade-offs between robustness and nominal quality. We then quantify the trade-offs with two measures "price of robustness" and "gain of robustness". In addition, we investigate how we can utilize the relationships between the three sets of solutions and the quantified trade-off to support decision making.

6.2 The Relationships between Three Sets of Robust Solutions

In our analysis in [PIV], we evaluate the set of point-based minmax robust Pareto optimal solutions X^{wc}, the set of nominal Pareto optimal solutions X^{nom}, and the set of lightly robust Pareto optimal solutions $X^{\text{light},\varepsilon}$ under the nominal case f^{nom} and the worst case f^{wc}. Then, we analyze the the relationships of different sets of solutions under the nominal scenario $f^{\text{nom}}(X^{\text{nom}})$, $f^{\text{nom}}(X^{\text{light},\varepsilon})$, and $f^{\text{nom}}(X^{\text{wc}})$ as well as the relationships of $f^{\text{wc}}(X^{\text{nom}})$, $f^{\text{wc}}(X^{\text{light},\varepsilon})$, and $f^{\text{wc}}(X^{\text{wc}})$ with the lower set less order and upper set-less order. The notations and set-based dominance are introduced in Chapter 3.

In Paper [PIV], we guarantee the domination property of f^{nom} and f^{wc} by some assumptions. Domination property means that a feasible solution is either Pareto optimal or there exists a Pareto optimal solution which dominates it. The formal definition of domination property is given in e.g., [40]. So, X^{nom} is a non-dominated set in f^{nom} and X^{wc} is a non-dominated set in f^{wc}. We summarize this property in Lemma 11:

Lemma 11. *For every set $\tilde{X} \subseteq X$ we have:*

(i) $f^{\text{nom}}(X^{\text{nom}}) \prec^{upp} f^{\text{nom}}(\tilde{X})$.
(ii) $f^{\text{wc}}(X^{\text{wc}}) \prec^{upp} f^{\text{wc}}(\tilde{X})$.

Under the nominal case f^{nom}, we find that $X^{\text{light},\varepsilon}$ dominates $f^{\text{nom}}(X^{\text{nom}}) + \varepsilon$ as presented in Lemma 12:

Lemma 12. *For every $\varepsilon \geq 0$ we have*

(i) $f^{\text{nom}}(X^{\text{light},\varepsilon}) \prec^{low} f^{\text{nom}}(X^{\text{nom}}) + \{\varepsilon\}$,
(ii) $f^{\text{nom}}(X^{\text{light},\varepsilon}) \prec^{upp} f^{\text{nom}}(X^{\text{nom}}) + \{\varepsilon\}$.

Under the worst case f^{wc}, we find that $X^{\text{light},\varepsilon}$ dominates X^{nom} and the lightly robust Pareto optimal solutions computed with a larger tolerance ε_2 dominates the set of lightly robust Pareto optimal solutions computed with a smaller tolerance ε_1 (here, we assumed the components of ε take the same value) as stated in Lemma 13:

Lemma 13.

(i) $f^{\text{wc}}(X^{\text{light},\varepsilon}) \prec^{upp} f^{\text{wc}}(X^{\text{nom}})$ *for all* $\varepsilon \geq 0$ *and*
(ii) $f^{\text{wc}}(X^{\text{light},\varepsilon_2}) \prec^{upp} f^{\text{wc}}(X^{\text{light},\varepsilon_1})$ *for all* $0 \leq \varepsilon_1 \leq \varepsilon_2$.

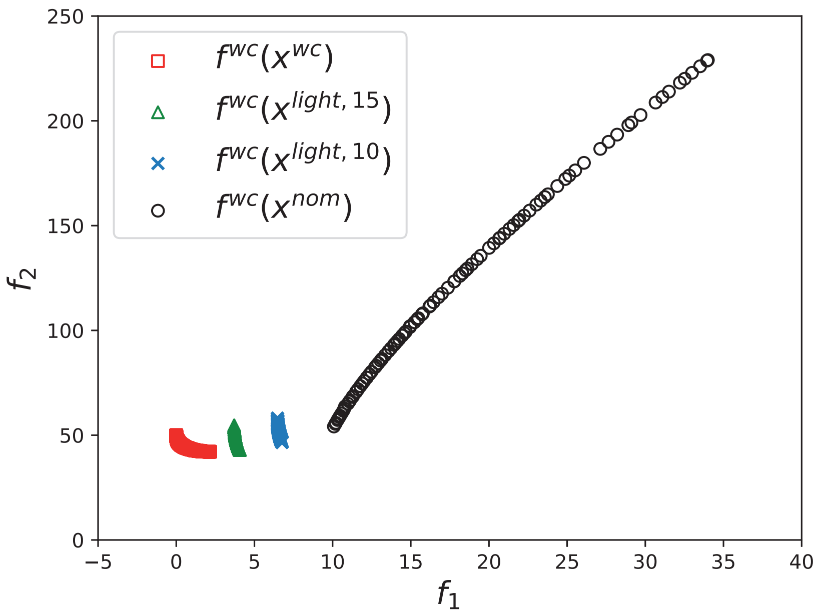


FIGURE 11 Relationships of different solution sets under f^{wc}

Combining the results of Lemma 11 and 13, we summarize our main findings concerning the relationships of the three sets of solutions under the worst case f^{wc}:

$$f^{\text{wc}}(X^{\text{wc}}) \prec^{upp} f^{\text{wc}}(X^{\text{light},\varepsilon_2}) \prec^{upp} f^{\text{wc}}(X^{\text{light},\varepsilon_1}) \prec^{upp} f^{\text{wc}}(X^{\text{nom}})$$

for $\varepsilon_1 \leq \varepsilon_2$. This means that choosing lightly robust Pareto optimal solutions result in better outcomes in the worst case than choosing the nominal Pareto optimal solutions. While the minmax robust Pareto optimal solutions are considered to be most "robust", lightly robust Pareto optimal solutions are "more robust" than the nominal Pareto optimal solutions.

This main finding is illustrated in Figure 11 for a bi-objective optimization problem with quadratic objective functions and an uncertain parameter in each objective function. In the figure, there are four different sets of solutions. We can see that $f^{\text{wc}}(X^{\text{wc}})$ dominates $f^{\text{wc}}(X^{\text{nom}})$ and $f^{\text{wc}}(X^{\text{light},\varepsilon})$ and $f^{\text{wc}}(X^{\text{nom}})$ is dominated by $f^{\text{wc}}(X^{\text{light},\varepsilon})$. As the two sets of lightly robust Pareto optimal solutions are concerned, we can see that the set of outcomes of $X^{\text{light},10}$ dominates the set of outcomes corresponding to $X^{\text{light},15}$. The purpose of using two values of ε is to demonstrate the finding that the set of lightly robust Pareto optimal solutions computed with larger ε dominates the set of lightly robust Pareto optimal solutions computed with smaller ε in the worst case. Using other values than 10 and 15, we would observe similar phenomena.

Combining the results of Lemma 11 and 12, we summarize our main findings concerning the relationships of the three sets of solutions in the nominal case f^{nom}:

$$f^{\text{nom}}(X^{\text{nom}}) \prec^{upp} f^{\text{nom}}(X^{\text{light},\varepsilon}) \prec f^{\text{nom}}(X^{\text{nom}}) + \{\varepsilon\}.$$

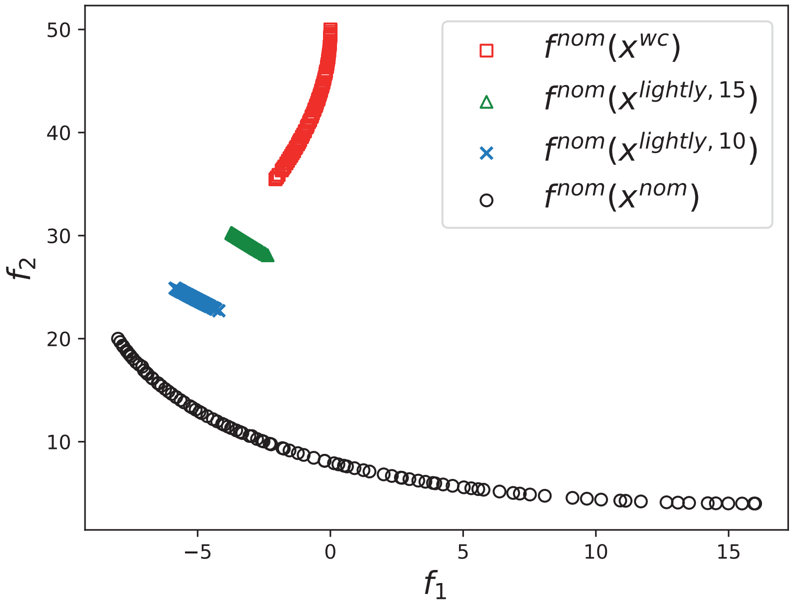


FIGURE 12 Relationships of different solution sets under f^{nom}

This means that in the nominal case, we can get a set of lightly robust Pareto optimal solutions with more similar outcomes to that of the set of nominal Pareto optimal solutions by decreasing ε.

This finding is visually illustrated in Figure 12 for the same problem as in Figure 11. In the figure, we can see that $f^{\text{nom}}(X^{\text{nom}})$ dominates $f^{\text{nom}}(X^{\text{light},\varepsilon})$. Note that in the figure we observe $f^{\text{nom}}(X^{\text{light},10})$ dominates $f^{\text{nom}}(X^{\text{light},15})$ and $f^{\text{nom}}(X^{\text{wc}})$. This relationship does not hold in general, as we show with a counterexample in Paper [PIV].

The relationships in f^{nom} and f^{wc} of the three sets of solutions provide a formal proof that lightly robust Pareto optimal solutions are good trade-offs between robustness and nominal quality. They also provide some guidelines on how to utilize the value of ε to adjust the robustness and nominal quality of the solutions in solving problem (11) to find lightly robust Pareto optimal solutions. Next, we quantify the trade-off between robustness and nominal quality, i.e., how much nominal quality is sacrificed to gain robustness and how much robustness we can gain. We call this the price of multiobjective robustness.

6.3 The Price of Multiobjective Robustness

With the price of multiobjective robustness, we measure the "price of robustness" and "gain of robustness". We first define the "price of robustness":

Definition 14. *Let $x \in X$ be a feasible solution to* (6). *We define its "price of robustness"* *as*

$$\text{price}(x) = \inf_{\bar{x} \in X^{\text{wc}}} \| f^{\text{nom}}(x) - f^{\text{nom}}(\bar{x}) \|_\infty,$$

where $\| \cdot \|_\infty$ *denotes the infinity-metric.*

The measure "price of robustness" describes how much nominal quality is given up to gain minmax robustness when comparing a feasible solution to a minmax robust Pareto optimal solution. Next, we specify "gain of robustness" to quantify how much better a minmax robust Pareto optimal solution $\bar{x}$ is in the worst case compared to a nominal Pareto optimal solution $\hat{x}$:

$$\text{gain}(\hat{x}, \bar{x}) = \| f^{\text{wc}}(\hat{x}) - f^{\text{wc}}(\bar{x}) \|_\infty.$$

6.4 Decision Support with The Price of Multiobjective Robustness

Next, we utilize the measures "price of robustness" and "gain of robustness" for decision support. We propose in Paper [PIV] two strategies the two-stage strategy and the lexicographic strategy. The two strategies are meant to support the decision maker in the choices of a nominal Pareto optimal or minmax robust Pareto optimal solution.

The Two-stage Strategy In the two-stage strategy, the decision maker first concentrates on the nominal scenario and identifies a most interesting nominal Pareto optimal solution $\hat{x}$. Then, we compute the "price of robustness" of the nominal Pareto optimal solution price$(\hat{x})$ by comparing to the nearest minmax robust Pareto optimal solution $\bar{x}$. The value of price$(\hat{x})$ is the price the decision maker needs to pay to change $\hat{x}$ to the nearest minmax robust Pareto optimal solution $\bar{x}$. We also compute "gain of robustness" gain$(\hat{x}, \bar{x})$, whose value describes how much better $\bar{x}$ is than $\hat{x}$ under the worst case.

With the knowledge of the values of price$(\hat{x})$ and gain$(\hat{x}, \bar{x})$, the decision maker can analyze if it is worthy of sacrificing the nominal quality to gain the amount of robustness. If it is, (s)he can choose the solution $\bar{x}$. If it is not, (s)he can choose the solution $\hat{x}$. In case (s)he wants to have a solution with a preferred trade-off between robustness and nominal quality, we need to compute a lightly robust Pareto optimal solutions for her or him.

The Lexicographic Strategy The lexicographic strategy is suitable when the decision maker does not have specific preferences on the objective function values. In this case, it is useful to find the nominal Pareto optimal solution with the smallest "price of robustness" or find the minmax robust Pareto optimal solution which is closest to the set of nominal Pareto optimal solutions. We can solve the following optimization problem to find a pair of solutions $\hat{x} \in X^{\text{nom}}$ and $\bar{x} \in X^{\text{wc}}$:

$$\min_{\hat{x} \in X^{\text{nom}}} \min_{\bar{x} \in X^{\text{wc}}} \| f^{\text{nom}}(\hat{x}) - f^{\text{nom}}(\bar{x}) \|_\infty. \tag{13}$$

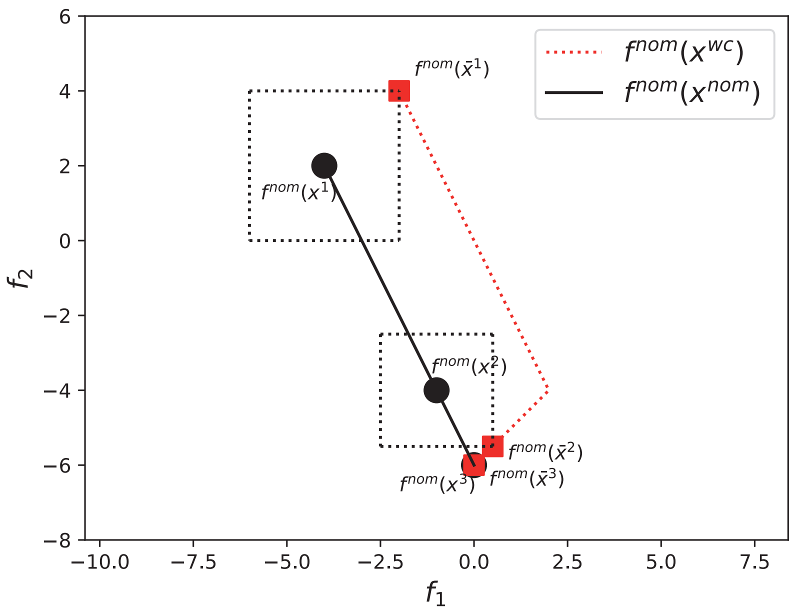


FIGURE 13 Nominal Pareto optimal solutions and their closest minmax robust Pareto optimal solutions.

Figure 13 illustrates examples of utilizing the two strategies in a simple linear problem. In the figure, the solution sets X^{nom} and X^{wc} are evaluated in the nominal case f^{nom}.

There are three nominal Pareto optimal solutions under consideration x^1, x^2, and x^3. For example, with the two-stage strategy, the decision maker first identifies x^2 as the most interesting solution, then we find the nearest minmax robust Pareto optimal solution $\bar{x}^2$. We compute $\text{price}(x^2) = 1.5$ and $\text{gain}(x^2, \bar{x}^2) = 1$ and present the values to the decision maker. The value of $\text{price}(x^2)$ represent the degradation in the nominal quality when solution x^2 is changed to the nearest minmax robust Pareto optimal solution $\bar{x}^2$. The value of $\text{gain}(x^2, \bar{x}^2)$ represents the improvement in the worst case outcome when x^2 is changed to $\bar{x}^2$. In the example, the objective functions do not have any meanings. In practical problems, the objectives have concrete meanings, so the decision maker can understand what the values mean in the problem context. Then, (s)he can compare the two values. In this case, based on the comparison on the numerical values, the decision maker may decide that it is not worthy of the sacrifice to gain robustness and choose x^2 as the final solution since the value of $\text{price}(x^2)$ is larger than $\text{gain}(x^2, \bar{x}^2)$.

With the lexicographic strategy, we consider two set of three solutions. We solve (13) and find a pair of nominal Pareto optimal and minmax robust Pareto optimal solutions closest to each other under f^{nom}. So, we find $\bar{x}^3$ in this case with $\text{price}(x^3) = 0$ and $\text{gain}(x^3, \bar{x}^3) = 0$. This solution is desirable for a decision maker who concentrates on nominal quality but wishes to gain robustness as much as possible. The solution is also good for a decision maker who wants to find a minmax robust Pareto optimal solution whose nominal quality is as good as possible.

The quantified "price of robustness" and "gain of robustness" provide concrete trade-off information for the decision maker to consider if it is worthy of sacrificing nominal quality to gain robustness. In addition, the two strategies can support the decision maker in finding a final solution. However, the computation of "price of robustness" and solving (13) in the lexicographic strategy requires pre-computed sets X^{nom} and X^{wc}. Without the pre-computed sets, it is hard to find their values.

The results presented in Paper [PIV] motivate us to develop an interactive multiobjective optimization method to support a decision maker to find a most preferred lightly robust Pareto optimal solution in Paper [PV]. In the next section, we summarize this paper.

7 BALANCING BETWEEN ROBUSTNESS AND NOMINAL QUALITY: DECISION SUPPORT WITH AN INTERACTIVE METHOD

7.1 Motivation

As discussed in Chapter 6, lightly robust Pareto optimal solutions are good trade-offs between robustness and nominal quality. The trade-offs between robustness and nominal quality can be adjusted by changing the value of ε in the lightly robust problem (11). The larger the value is, the closer the lightly robust Pareto optimal solutions are to the minmax robust Pareto optimal solutions. This means that the more "robust" the lightly robust Pareto optimal solutions are.

In addition, we also proposed two strategies to support the decision maker to choose between nominal Pareto optimal solutions and minmax robust Pareto optimal solutions based on the measures of "price of robustness" and "gain of robustness". In case that a decision maker wants to find a solution with preferred trade-offs between robustness and nominal quality, we can support the decision maker to find a most preferred lightly robust Pareto optimal solution. By a most preferred lightly robust Pareto optimal solution we mean that 1) the robustness and nominal quality are balanced based on the decision maker's preference and 2) the nominal objective function values satisfy the preferences of the decision maker as well as possible.

As mentioned earlier, the lightly robust problem (11) requires us to fix a nominal Pareto optimal solution first. This requirement allows us to incorporate the decision maker's preferences on the nominal outcome. In addition, the decision maker can specify the value of ε which represents how much nominal quality (s)he is willing to sacrifice to gain robustness. Based on this idea, we develop a lightly robust interactive multiobjective optimization (LiRoMo) method in Paper [PV]. We summarize LiRoMo in this chapter.

7.2 Building the LiRoMo Method

In LiRoMo, the decision maker is expected to specify the following preference information:

1. A reference point, which consists of desired nominal objective function values.
2. The tolerable degradation ε on the nominal objective function values.

Based on the preferences, we first find a nominal Pareto optimal solution $\hat{x}$ which satisfies the reference point as well as possible by solving (3). Then, we find a lightly robust Pareto optimal solution $x^{\text{light},\varepsilon}$ with respect to $\hat{x}$ and ε by solving the following lightly robust subproblem based on the achievement scalarizing function:

$$
\begin{array}{ll}
\text{minimize} & \alpha + \rho \sum_{i=1}^{k} (\max_{\xi \in \mathcal{U}} f(x, \xi) - \bar{z}_i) \\
\text{subject to} & f_i(x, \hat{\xi}) \leq f_i(\hat{x}, \hat{\xi}) + \varepsilon_i \text{ for all } i = 1, \cdots, k \\
 & w_i(\max_{\xi \in \mathcal{U}} f_i(x, \xi) - \bar{z}_i) \leq \alpha \text{ for all } i = 1, \cdots, k \\
 & x \in X.
\end{array} \tag{14}
$$

In Paper [PV], we prove that an optimal solution of (14) is a lightly robust Pareto optimal solution to (6). In addition, we reformulate (14) under some assumptions so that it can be efficiently solved. We considered the reformulation under polyhedral uncertainty set with quasi convex objective functions with respect to the uncertain parameters. As a special case of polyhedral uncertainty, interval uncertainty is included in the reformulation with quasi convex objective functions. The reformulation is also valid for uncertainty sets consisting of a set of discrete scenarios without the assumption on quasi convexity of the objective functions.

In order to support the decision maker to make informed decisions and find a preferred balance between robustness and nominal quality, we use the measures "price to be paid for robustness" and "gain of robustness" which originate from [PIV]. We used the term "price for robustness" in Chapter 6. Here the term "price to be paid for robustness" carries the message that it is the price to be paid only if the lightly robust Pareto optimal solution is chosen. Since the objective functions can be with very different magnitudes, we should normalize the two measures such that they are more understandable for the decision maker as follows:

$$
\text{gain}(x^{\text{light},\varepsilon}, \hat{x}) = \left\| \left(\frac{f_1^{\text{wc}}(x^{\text{light},\varepsilon}) - f_1^{\text{wc}}(\hat{x})}{z_1^{\text{nad,wc}} - z_1^{\text{uto,wc}}}, \ldots, \frac{f_k^{\text{wc}}(x^{\text{light},\varepsilon}) - f_k^{\text{wc}}(\hat{x})}{z_k^{\text{nad,wc}} - z_k^{\text{uto,wc}}} \right)^T \right\|_\infty,
$$

and

$$
\text{price}(x^{\text{light},\varepsilon}, \hat{x}) = \left\| \left(\frac{f_1^{\text{nom}}(x^{\text{light},\varepsilon}) - f_1^{\text{nom}}(\hat{x})}{z_1^{\text{nad}} - z_1^{\text{uto}}}, \ldots, \frac{f_k^{\text{nom}}(x^{\text{light},\varepsilon}) - f_k^{\text{nom}}(\hat{x})}{z_k^{\text{nad}} - z_k^{\text{uto}}} \right)^T \right\|_\infty.
$$

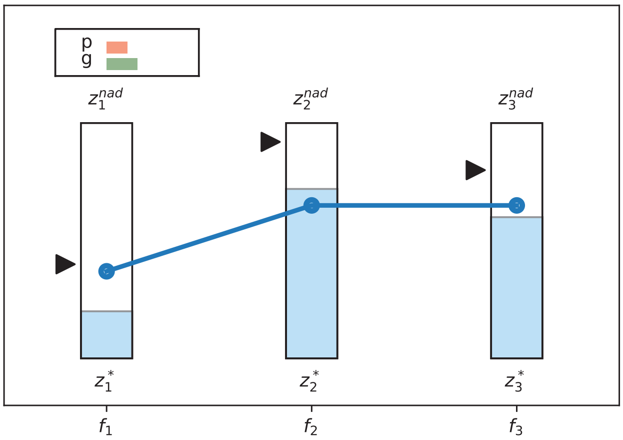


FIGURE 14 Visualizing a lightly robust Pareto optimal solution

The measure gain$(x^{\text{light},\varepsilon}, \hat{x})$ quantifies how much better the lightly robust Pareto optimal solution is compared to the nominal Pareto optimal solution in the worst case, i.e., how much "robustness" can be gained. The measure price$(x^{\text{light},\varepsilon}, \hat{x})$ quantifies how much worse the lightly robust Pareto optimal solution is compared to the nominal Pareto optimal solution in the nominal case, i.e., how much nominal quality is to be sacrificed.

In order to visually support the decision maker to process the information related to the solutions, we visualize the solution with an augmented value path as shown in Figure 14. In the figure, the decision maker can observe five different kinds of information:

1. The nominal objective function values of the current nominal Pareto optimal solution which satisfies the reference point best in the colored bars. The colored bars are within the white bars which represent the ranges of the objective function values.
2. The nominal objective function values of the current lightly robust Pareto optimal solution.
3. The change in the nominal quality, i.e., the difference between the markers of the value path and the colored part of the bars.
4. How much better the current lightly robust Pareto optimal solution is compared to the worst acceptable one whose nominal objective function values are marked by the triangles in the figure.
5. How much robustness (s)he can gain compared to the sacrificed nominal quality. The information is only presented with two small colored bars marked by p (for price) g (for gain). The decision maker can compare their lengths to get sufficient information without the numerical values. If the bar marked by p is longer than the bar marked by g, more nominal quality is sacrificed compared to robustness to be gained. If the situation is opposite, less nominal quality is sacrificed compared to robustness to be gained. If the two bars are similar length, the gain and price are similar.

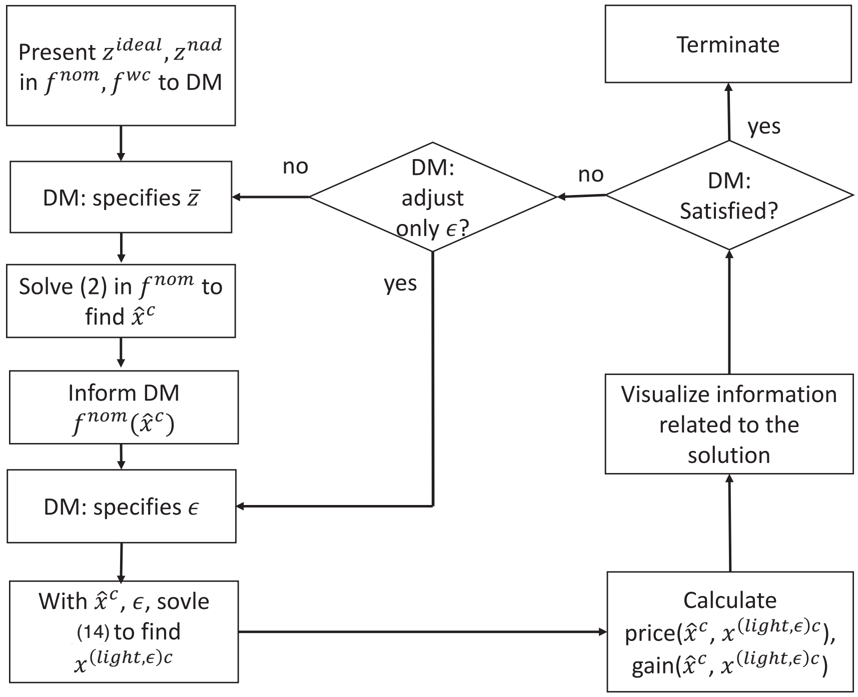


FIGURE 15 The steps of LiRoMo

7.3 LiRoMo

The LiRoMo method supports the decision maker in updating her or his preferences iteratively and directs the solution process towards a most preferred lightly robust Pareto optimal solution. The steps of the LiRoMo method are illustrated in Figure 15.

As shown in the figure, the basic idea of LiRoMo is to ask for the decision maker's preferences (a reference point) on the nominal outcome. And, find a nominal solution which satisfies this preferences first. Then the decision maker is informed about the nominal outcome of this solution. With respect to the information, the decision maker is expected to specify how much nominal quality (s)he is willing to sacrifice. Based on the nominal solution and the preferences on ε, we calculate a lightly robust Pareto optimal solution and visually present related information to the decision maker (like in Figure 14). If the decision maker wants to alter the value of ε, we find a new lightly robust Pareto optimal solution. If the decision maker wishes to start with another nominal Pareto optimal solution, (s)he is expected to specify a new reference point. This kind of steps carries on until the decision maker finds a lightly robust Pareto optimal solution with satisfactory nominal outcome and good trade-off between nominal quality and robustness.

For demonstrating the application of LiRoMo to support decision making, we formulated and solved a stock investment portfolio optimization problem for a small sized start-up to find a lightly robust investment portfolio. The objectives were maximizing the return on investment and minimizing risks (via multiple risk measures). The uncertainty in their investments came from the fact that they did not know exactly how long their investments would stay in the stock market. They planed for a long term investment. But, if they need the invested money to start new projects, the investment would be withdraw earlier. So, we considered three scenarios simultaneously including the investment as a long term investment, a mid-term investment and a short term investment. The different investment periods require using different data, which is reflected as parameters in the objective functions.

With the LiRoMo method, we were able to support the decision maker to consider the planned long term investment case and the earlier withdraw cases simultaneously. As a result, the decision maker found an investment portfolio with a good long term return and acceptable risks. At the same time, the investment portfolio was not too bad in case of earlier withdrawal. The development of the LiRoMo method resulted in the first interactive method for lightly robust Pareto optimal solutions. LiRoMo also answered the question of how to support the decision maker to make informed decisions and find a good balance between robustness and nominal quality.

8 DISCUSSIONS

In this section, we first discuss some insights gained in the development of this thesis. Then we derive some suggestions for applying the developed methods. Some of them are either explicitly or implicitly mentioned in the previous chapters. We deepen the discussion here since they are very important elements to consider when solving practical problems.

8.1 Experimenting on Example Problems with Decision Makers

In our experiments, when solving the example problems presented in the included papers, the interactions are done with real decision makers. Not only the authors of the papers but also some colleges and friends acted as decision makers. We carried out multiple experiments during the developments of the interactive methods. The feedbacks helped us to refine our design of the methods. For each example problem, we reported one of the interactive solution processes among many of them.

We say that we terminate the solution process when the decision maker finds a most preferred or satisfied solution. The methods and strategies for measuring the satisfaction are not at all trivial. In our experiments, we relied on the statements and opinions of the decision makers. Better assessment on the interactive methods include for example repeating the experiments with a wider range of decision makers and use questionnaires or interviews to access their satisfaction. Alternatively, we can also evaluate the interactive methods based on simulations, e.g., using artificial decision makers [65].

In addition, the interactive methods developed in this thesis have not been tested in different problem domains. These methods still await more applications in different disciplines. Solving different problems will provide feedbacks from different view. The feedbacks should be utilized in refining the methods as well as developing new features.

8.2 Visualizing A Solution

In this thesis, we developed different kinds of visualizations. The visualizations are used to support decision makers in understanding related information of the solutions. Visualization is important because it can "provide extensions to mental process" [17], thus, help the decision makers in understanding the solutions. During the development of the visualization, we have explored other possibilities. For example, we have experimented on augmenting the pedal diagrams and different kinds of bar charts (see e.g., [57] for a summary of these visualization techniques). The visualizations we chose serve the purpose of delivering different kinds of information related to robustness and objective function values in an understandable way to decision makers.

Typically, in our experiments, we explained the visualizations to the decision makers. It is an important action before presenting solutions via the visualizations. When we have more information for the decision maker to consider, it takes time for the decision maker to train. Typically, human can only handle limited amount of information. For example, when there are a large number of objectives, the visualizations developed in this thesis can contain too much information for a decision maker to comprehend. In this situation, the visualization of information on a single objective is not affected. But, grasping an overview can be challenging. Thus, There should be a cognitive upper bound of the decision maker. In this case, the trade-off has to be considered between how much information the decision maker can digest and information about how many different objectives or aspects we should provide to her or him.

In addition, other issues like the arrangements of visual elements and color usage were not within the scope of this thesis. These aspects can affect the actual decisions of the decision makers (see, e.g., [47]). So, they should be considered when designing user interfaces for the interactive methods.

8.3 Recognizing Uncertainty

Sometimes, it can be challenging to recognize the sources of uncertainty. As mentioned earlier, it can also be challenging for the decision maker to estimate the uncertainty sets for imprecise information or uncertain future developments. We assumed that decision makers can provide information about the uncertainty sets by e.g., utilizing her or his own domain knowledge or using data from the past. If a decision maker is not confident in providing such information, we have to utlize some other techniques to construct the uncertainty set, e.g., using machine learning techniques (see e.g., [77]).

8.4 Choice of A Robustness Concept

When solving practical problems, we face the challenge of choosing a robustness concept to apply. Subsequently, when we consider supporting a decision maker, we need to handle the choice of an interactive method. The interactive methods developed in this thesis will be available in the DESDEO framework [1]. In this thesis, we assumed that the problem formulation reflects the real problem to be solved. So, the discussion here is focused on the practical consequences and the meaning for the decision maker of using different robustness concepts considered in this thesis.

When we use a robustness measure, we focus on quantifying the effects of uncertainty in the objective functions. In this kind of approach, robustness means how the objective function values vary due to uncertainty. A robustness measure is suitable for problems where the values of uncertain parameters/decision variables are perturbed in a small neighborhood of their nominal values. If the uncertainty is reflected in a different way, the ranges of variation can be so big that it does not help in decision making. Also, the measure cannot be used alone, it should be combined with the objectives because the focus of the decision maker is still on the objectives. Introducing robustness with a measure is to guarantee that the objective function values do not deviate as unexpected.

When there are a set of possible values of uncertain parameters which are not any small perturbations in the neighborhood of the nominal values, it can be more suitable to adopt some robustness concepts. When the decision maker is e.g., risk averse, we can use minmax robustness to help her or him to find a solution which is still valid even when the worst case occurs. Depending on how the decision maker wishes to make decisions, we can for example either use the robust ASF approach or the MuRO-NIMBUS method. The robust ASF approach can support the decision maker to find a solution with the most preferred worst case outcome. MuRO-NIMBUS can support the decision maker to find a solution corresponding to a most preferred nominal outcome. The solutions found provide a pessimistic view on the consequences of uncertainty involved. The decision maker might miss some better solutions in terms of outcomes in other scenarios and her or his preferences.

When the decision maker is e.g., risk-seeking, we can solve the problem in the nominal scenario. We do not consider the "best case scenarios" here, because if the outcomes of solutions are better than expected, it can be a positive surprise for the decision maker. When we support a e.g., risk-neutral decision maker who wants to consider both robustness and nominal quality simultaneously. We can use light robustness and apply the LiRoMo method. When the decision maker can find a satisfactory trade-off between the two aspects, (s)he also needs to deal with more information in terms of both understanding the solutions and specifying preferences. In general, the choice of a robustness concept can depend on the problem itself as well as the type of solution the decision maker desires.

9 AUTHOR'S OWN CONTRIBUTION

The research topic of utilizing interactive methods in multiobjective robust optimization was proposed by the author's supervisors. In the beginning, the author reviewed the literature and found that the consideration should be in at least two kinds of uncertainty: decision uncertainty and parameter uncertainty.

After some discussions, the supervisors and the author decided together to start by considering decision uncertainty and using a robustness measure to quantify the effects of uncertainty. The author investigated how she can incorporate a robustness measure to an interactive method. After finding the desired characteristics of a robustness measure, the author discovered that none of the existing robustness measures met the needs of supporting decision making in an interactive method. So, the author developed the proposed robustness measure. The robustness measure was matured based on some feedbacks from the supervisors. The modification of the NIMBUS method was done by the author, and the visualization was also developed by the author. The author constructed the example problems and conducted the numerical experiments. In addition, the author was responsible for writing major parts of the text of Paper [PI]. The supervisors also helped the author to improve the manuscript of Paper [PI].

During Prof. Schöbel's visit in Jyväskylä, the author generated the idea of Paper [PII] based on discussions with Prof. Schöbel and the supervisors. The author developed the MuRO-NIMBUS method. The author also constructed the example problem and conducted all the numerical experiments. The author presented the research in the SimScience workshop. The author wrote major parts of the paper. The supervisors commented on the manuscript. The comments helped the author to improve Paper [PII].

When the author was developing the MuRO-NIMBUS method, she realized that solving the scalarized subproblem was challenging. She described this issue to her colleagues and got the inspiration of seeking for some new ideas from the evolutionary multiobjective optimization field. After some investigation, the author generated the idea in Paper [PIII]. The author extended an implementation of SIBEA for the SIBEA-R method. She implemented new modules and integrated them to the SIBEA implementation. The author conducted all the numerical experiments and wrote the text in Paper [PIII]. Prof. Miettinen commented on the text of Paper [PIII], which helped the author to improve the manuscript.

During the author's visit in Prof. Schöbel's research group, she talked with Prof. Schöbel about her ideas of working on light robustness. Her intuitions on the relationships between lightly robust, nominal, and minmax robust Pareto optimal solutions were also communicated. Prof. Schöbel suggested to analyze their relationships using set-based dominance. Prof. Schöbel formalized the proofs in Paper [PIV] while the author conducted the numerical experiments. The measures "price of robustness" and "gain of robustness" were developed as a joint effort. The two-stage strategy was the author's idea while the lexicographic strategy was Prof. Schöbel's idea. As a result, the author wrote the texts of Paper [PIV] together with Prof. Schöbel.

With the solid basis that lightly robust Pareto optimal solutions are good trade-offs between robustness and nominal quality, the author is motivated to develop the LiRoMo method. The development of the LiRoMo method was done by the author. The discussions with the supervisors helped to mature the ideas during the developments. The author formulated the example problem and conducted the numerical experiments. The author wrote the text of Paper [PV]. Prof. Miettinen's comments helped to improve the manuscript.

Overall, the supervisors have guided the author more intensively at the beginning of the research. In the later stage, the author was the primary source of new ideas. In the whole process of preparing the thesis, the author was the main source of developing the interactive methods. The author also constructed the example problems and conducted the numerical experiments. Feedbacks and comments from the supervisors and Prof. Schöbel were useful for the author to consider various aspects in developing interactive methods for multiobjective robust optimization.

10 CONCLUSIONS AND FUTURE RESEARCH DIRECTIONS

The goal of this thesis is to develop interactive methods for tackling various challenges involved in supporting decision makers to make informed decisions for practical multiobjective optimization problems under uncertainty. We utilized the multiobjective robust optimization approach. In the literature, the research efforts have been mainly devoted to developments of different robustness concepts and discussions on their practical applicability in decision making are hard to find. This thesis analyzed different robustness measures and concepts from the perspective of their practical applicability in decision making. In addition, the interactive methods developed in this thesis provide the needed tools for supporting decision makers in multiobjective robust optimization.

We first concentrated on multiobjective optimization problems with decision uncertainty. We proposed a new robustness measure and incorporated it into the NIMBUS method as an example of using a robustness measure in an interactive method. For multiobjective optimization problems with parameter uncertainty, we first considered supporting a decision maker to find a most preferred minmax robust Pareto optimal solution based on their outcomes in the nominal scenario. As a result, we developed the MuRO-NIMBUS method which involved the pre-decision making and the decision making stage. Since it is very challenging to compute minmax robust Pareto optimal solutions in the pre-decision making stage, we utilized ideas from evolutionary multiobjective optimization and developed the SIBEA-R method.

With the development of MuRO-NIMBUS, we realized that, sometimes, finding minmax robust Pareto optimal solutions is not enough. There can be cases that a decision maker prefers to search for a satisfactory trade-off between robustness and quality. After the exploration for suitable concepts to introduce the trade-off, we analyzed nominal Pareto optimal solutions, minmax robust Pareto optimal solutions, and lightly robust Pareto optimal solutions. Based on our analysis, we found that lightly robust Pareto optimal solutions are good trade-offs between robustness and quality. We quantified the trade-offs with "price of robustness" and "gain of robustness".

For supporting decision makers to find a preferred trade-off between robustness and quality, we developed the LiRoMo method. In LiRoMo, the decision maker can affect the nominal objective function values (i.e., nominal quality) of the lightly robust Pareto optimal solutions. At the same time, the decision maker can control the trade-off between robustness and nominal quality.

As mentioned before, the overall goal of this research is to support decision makers in making informed decisions for multiobjective optimization problems under uncertainty. As a result, decision makers are aware of and being well prepared for involved uncertainty. This thesis presents achievements in some aspects by bridging the theoretical developments and practical decision making in multiobjective robust optimization. For achieving the overall goal, there are still many interesting future research venues.

For supporting the decision maker with sufficient information, we utilize sets of pre-computed solutions. For example in MuRO-NIMBUS, we computed a set of set-based minmax robust Pareto optimal solutions to help the decision maker prepare for the worst case. Also, we need a set of point-based minmax robust Pareto optimal solution in our two-stage and lexicographic strategies. Developing approximation methods to construct approximated sets of robust Pareto optimal solutions is an interesting topic. The results from this line of research will help to improve the effectiveness of the developed methods and strategies and consequently enable better support for decision makers.

The developed interactive methods involves scalarized subproblems. The scalarized subproblems are very challenging to solve. Solving the subproblems is an important element in the interactive methods. We considered reformulation of problems with some assumed characteristics. Thus, reformulations for wider class of problems is also a relevant future research direction. The results from this research direction enables us to solve the subproblems more efficiently.

In addition, preservation of the decision maker's preferences in different scenarios (e.g., when considering the nominal scenario and the worst case) is also a valid future research topic. In this research topic, the aim should be at finding solutions which reflect decision makers' preferences in different scenarios as well as possible. With this kind of solution, the decision makers' preferences in different point of view can be satisfied.

As discussed before, the interactive methods proposed in this thesis still await more applications in different disciplines. For different applications, fine-tuning the methods for some specific application problems can be necessary. Feedbacks from decision makers in different problem domain can facilitate the developments of new features. Different application problems can also motivate us to explore possibilities in other robustness concepts and other types of interactive methods. For example, extending the concept of adjustable robustness [9] to multiobjective optimization problems can be beneficial for forest treatment planning where decisions are made for a long time horizon involving different kinds of uncertainty.

To summarize, with the analysis on the practical applicability of robustness measures and concepts in decision making, we bridge the theory of multiobjective robust optimization and the practice of decision making in solving multiobjective optimization problems under uncertainty. The developed interactive methods provide necessary tools to support decision making under different kinds of uncertainty and find different kinds of solutions. Together, the results of analysis and the developed interactive methods can provide some insights and guidelines for researchers and practitioners who consider solving multiobjective optimization problems under uncertainty. Overall, this thesis brings multiobjective robust optimization into practice.

YHTEENVETO (FINNISH SUMMARY)

Interaktiivisia menetelmiä robustiin monitavoiteoptimointiin

Käytännön optimointiongelmissa on tyypillisesti useita ristiriitaisia tavoitteita, joissa on eri lähteistä syntyvää epävarmuutta. Erilaisia robustisuuden käsitteitä on kehitetty huomioimaan useat tavoitteet ja niiden epävarmuus samanaikaisesti. Kuitenkaan niiden käytännön toimivuutta ei ole juurikaan kirjallisuudessa käsitelty. Vielä vähemmän on tutkittu ratkaisumenetelmiä, joilla voidaan tukea päätöksentekijää löytämään paras robusti ratkaisu, joka ei ole liian herkkä epävarmuuksille. Siksi tässä väitöskirjatutkimuksessa keskitytään kahteen pääteemaan: erilaisten robustisuuskäsitteiden käytännön soveltuvuuteen päätöksenteossa ja interaktiivisten eli vuorovaikutteisten menetelmien kehittämiseen, jotta päätöksentekijää voidaan tukea löytämään häntä parhaiten tyydyttävä robusti ratkaisu erilaisten epävarmuuksien vallitessa ristiriitaisten tavoitteiden välillä.

Työssä käsitellään aluksi tilanteita, joissa optimoitujen ratkaisujen käytännön toteutusta ei voida taata tarkasti, vaan toteutuksessa on epävarmuutta. Tällaisille ongelmille kehitetään uusi robustisuusmittari, jolla epävarmuuden vaikutusta optimoitavien funktioiden arvoihin voidaan mitata. Tämä mittari kytketään osaksi interaktiivista menetelmää, jossa päätösentekijälle esitetään ratkaisuja ja tietoa niiden robustisuudesta tehostettujen havainnollistuskeinojen avulla.

Työssä käsitellään myös epävarmuutta optimoitavien funktioiden parametreissa. Ensin käytetään joukkopohjaista min-max-robustisuuskäsitettä ja kehitetään kaksivaiheinen interaktiivinen menetelmä tukemaan päätöksentekijää löytämään hänelle mieluisin ns. joukkopohjainen min-max-robusti, Pareto-optimaalinen ratkaisu. Koska tällaisia ratkaisuja on vaikeaa laskea, kehitetään niiden approksimointiin evoluutiopohjainen monitavoiteoptimoinnin menetelmä. Tämän jälkeen analysoidaan erilaisia robustisuuskäsitteitä ja todetaan, että kevyesti robustit Pareto-optimaaliset ratkaisut tarjoavat hyvän tasapainon robustisuuden ja tyydyttävien optimoitavien funktioiden arvojen välille. Uusi menetelmä kehitetään tukemaan päätöksentekijää löytämään paras kevyesti robusti Pareto-optimaa linen ratkaisu.

Väitöskirjan tulokset laajentavat robustisuuskäsitteiden soveltuvuutta käytännön päätöksenteossa. Lisäksi kehitetyt uudet menetelmät vievät robustia monitavoiteoptimointia käytännön päätöksenteon tueksi.

REFERENCES

[1] Decision support for computationally demanding optimization problems. https://desdeo.it.jyu.fi.

[2] J. Allen, C. Seepersad, H. Choi, and F. Mistree. Robust design for multiscale and multidisciplinary applications. *Journal of Mechanical Design*, 128(4):832–843, 2006.

[3] M. Asafuddoula, H. Singh, and T. Ray. Six-sigma robust design optimization using a many-objective decomposition-based evolutionary algorithm. *IEEE Transactions on Evolutionary Computation*, 19(4):490–507, 2015.

[4] G. Avigad and J. Branke. Embedded evolutionary multi-objective optimization for worst case robustness. In M. Keijzer, editor, *Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation*, pages 771–778, 2008.

[5] J. Bader, D. Brockhoff, S. Welten, and E. Zitzler. On using populations of sets in multiobjective optimization. In M. Ehrgott, C. M. Fonseca, X. Gandibleux, J. K. Hao, and M. Sevaux, editors, *Evolutionary Multi-Criterion Optimization, Proceedings*, pages 140–154. Springer, 2009.

[6] C. Barrico and C. Antunes. Robustness analysis in multi-objective optimization using a degree of robustness concept. In *Proceedings of the IEEE Congress on Evolutionary Computation, CEC2006*, pages 1887–1892, 2006.

[7] A. Ben-Tal, D. Bertsimas, and D. B. Brown. A soft robust model for optimization under ambiguity. *Operations Research*, 58(4-part-2):1220–1234, 2010.

[8] A. Ben-Tal, L. El Ghaoui, and A. Nemirovski. *Robust Optimization*. Princeton University Press, 2009.

[9] A. Ben-Tal, A. Goryashko, E. Guslitzer, and A. Nemirovski. Adjustable robust solutions of uncertain linear programs. *Mathematical Programming*, 99(2):351–376, 2004.

[10] D. Bertsimas, D. B. Brown, and C. Caramanis. Theory and applications of robust optimization. *SIAM Review*, 53(3):464–501, 2011.

[11] D. Bertsimas and M. Sim. Robust discrete optimization and network flows. *Mathematical Programming*, 98(1):49–71, 2003.

[12] D. Bertsimas and M. Sim. The price of robustness. *Operations Research*, 52(1):35–53, 2004.

[13] J. R. Birge and F. Louveaux. *Introduction to Stochastic Programming*. Springer, 2011.

[14] G. R. Bitran. Linear multiple objective problems with interval coefficients. *Management Science*, 26(7):694–706, 1980.

[15] R. Bokrantz and A. Fredriksson. Necessary and sufficient conditions for Pareto efficiency in robust multiobjective optimization. *European Journal of Operational Research*, 262(2):682–692, 2017.

[16] J. Branke, K. Deb, K. Miettinen, and R. Slowinski, editors. *Multiobjective Optimization, Interactive and Evolutionary Approaches*. Springer, 2008.

[17] K. W. Brodlie, L. A. Carpentier, R. A. Earnshaw, J. R. Gallop, R. J. Hubbold, A. M. Mumford, C. D. Osland, and P. Quarendon, editors. *Scientific Visualization: Techniques and Applications*. Springer, 1992.

[18] V. Chankong and Y. Y. Haimes. Optimization-based methods for multiobjective decision-making: An overview. *Large Scale Systems*, 5(1):1–23, 1983.

[19] R. Cheng, Y. Jin, M. Olhofer, and B. Sendhoff. A reference vector guided evolutionary algorithm for many-objective optimization. *IEEE Transactions on Evolutionary Computation*, 20(5):773–791, 2016.

[20] C. A. C. Coello, G. B. Lamont, and D. A. V. Veldhuizen. *Evolutionary Algorithms for Solving Multi-Objective Problems, 2nd edition*. Springer, 2006.

[21] K. Deb. *Multi-Objective Optimization Using Evolutionary Algorithms*. John Wiley & Sons, Inc., 2001.

[22] K. Deb and H. Gupta. Introducing robustness in multi-objective optimization. *Evolutionary Computation*, 14(4):463–494, 2006.

[23] K. Deb, K. Miettinen, and S. Chaudhuri. Toward an estimation of nadir objective vector using a hybrid of evolutionary and local search approaches. *IEEE Transactions on Evolutionary Computation*, 14(6):821–841, 2010.

[24] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.

[25] G. M. Dranichak and M. M. Wiecek. On highly robust efficient solutions to uncertain multiobjective linear programs. *European Journal of Operational Research*, 2018. doi: 110.1016/j.ejor.2018.07.035.

[26] M. Ehrgott, J. Ide, and A. Schöbel. Minmax robustness for multi-objective optimization problems. *European Journal of Operational Research*, 239(1):17–31, 2014.

[27] G. Eichfelder, C. Krüger, and A. Schöbel. Decision uncertainty in multiobjective optimization. *Journal of Global Optimization*, 69(2):485–510, 2017.

[28] M. Fischetti and M. Monaci. Light robustness. In R. K. Ahuja, R. H. Möhring, and C. D. Zaroliagis, editors, *Robust and Online Large-Scale Optimization: Models and Techniques for Transportation Systems*, pages 61–84. Springer, 2009.

[29] J. Fliege and R. Werner. Robust multiobjective optimization & applications in portfolio optimization. *European Journal of Operational Research*, 234(2):422–433, 2014.

[30] A. Gaspar-Cunha and J. A. Covas. Robustness in multi-objective optimization using evolutionary algorithms. *Computational Optimization and Applications*, 39(1):75–96, 2007.

[31] A. M. Geoffrion, J. S. Dyer, and A. Feinberg. An interactive approach for multi-criterion optimization, with an application to the operation of an academic department. *Management Science*, 19(4-part-1):357–368, 1972.

[32] P. G. Georgiev, D. T. Luc, and P. M. Pardalos. Robust aspects of solutions in deterministic multiple objective linear programming. *European Journal of Operational Research*, 229(1):29–36, 2013.

[33] M. Goberna, V. Jeyakumar, G. Li, and J. Vicente-Pérez. Robust solutions to multi-objective linear programs with uncertain data. *European Journal of Operational Research*, 242(3):730 – 743, 2015.

[34] M. Goberna, V. Jeyakumar, G. Li, and J. Vicente-Pérez. Guaranteeing highly robust weakly efficient solutions for uncertain multi-objective convex programs. *European Journal of Operational Research*, 270(1):40–50, 2018.

[35] M. Goerigk and A. Schöbel. Algorithm engineering in robust optimization. In L. Kliemann and P. Sanders, editors, *Algorithm Engineering: Selected Results and Surveys*, pages 245–279. Springer, 2016.

[36] S. Gunawan and S. Azarm. Multi-objective robust optimization using a sensitivity region concept. *Structural and Multidisciplinary Optimization*, 29(1):50–60, 2005.

[37] D. Hadka, J. Herman, P. Reed, and K. Keller. An open source framework for many-objective robust decision making. *Environmental Modelling & Software*, 74:114–129, 2015.

[38] F. Hassanzadeh, H. Nemati, and M. Sun. Robust optimization for multiobjective programming problems with imprecise information. *Procedia Computer Science*, 17:357–364, 2013.

[39] F. Hassanzadeh, H. Nemati, and M. Sun. 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.

[40] M. I. Henig. The domination property in multicriteria optimization. *Journal of Mathematical Analysis and Applications*, 114(1):7–16, 1986.

[41] Z. Huan, G. Zhenghong, X. Fang, and Z. Yidian. Review of robust aerodynamic design optimization for air vehicles. *Archives of Computational Methods in Engineering*, 2018.

[42] C. L. Hwang and A. S. M. Masud. Multiple objective decision making methods and applications: A state-of-the-art survey. In *Lecture Notes in Economics and Mathematical Systems*, volume 164. Springer, 1979.

[43] J. Ide, E. Köbis, D. Kuroiwa, A. Schöbel, and C. Tammer. The relationship between multi-objective robustness concepts and set-valued optimization. *Fixed Point Theory and Applications*, 2014(1):83, 2014.

[44] J. Ide and A. Schöbel. Robustness for uncertain multi-objective optimization: a survey and analysis of different concepts. *OR Spectrum*, 38(1):235–271, 2016.

[45] J. Ide, M. Tiedemann, S. Westphal, and F. Haiduk. An application of deterministic and robust optimization in the wood cutting industry. *4OR*, 13(1):35–57, 2015.

[46] M. Inuiguchi, H. Ichihashi, and H. Tanaka. Fuzzy programming: A survey of recent developments. In S. Y. Huang and J. Teghen, editors, *Stochastic Versus Fuzzy Approaches to Multiobjective Mathematical Programming under Uncertainty*, pages 45–68, 1990.

[47] S. L. Jarvenpaa. The effect of task demands and graphical format on information processing strategies. *Management Science*, 35(3):285–303, 1989.

[48] J. R. Kasprzyk, S. Nataraj, P. M. Reed, and R. J. Lempert. Many objective robust decision making for complex environmental systems undergoing change. *Environmental Modeling & Software*, 42:55–71, 2013.

[49] A. Khan, C. Tammer, and C. Zălinescu. *Set-valued Optimization*. Springer, 2015.

[50] D. Konur and H. Farhangi. Set-based min-max and min-min robustness for multiobjective robust optimization. In K. Coperich, E. Cudney, and H. Hembhard, editors, *Proceedings of the 2017 Industrial and Systmes Engineering Research Conference*, pages 1–6. Institute of Industrial and Systems Engineers, 2017.

[51] P. J. Korhonen and J. Laakso. A visual interactive method for solving the multiple criteria problem. *European Journal of Operational Research*, 24(2):277–287, 1986.

[52] K. Kuhn, A. Raith, M. Schmidt, and A. Schöbel. Bi-objective robust optimisation. *European Journal of Operational Research*, 252(2):418–431, 2016.

[53] D. Kuroiwa and G. M. Lee. On robust multiobjective optimization. *Vietnam Journal of Mathematics*, 40(2-3):305–317, 2012.

[54] M. Li, S. Azarm, and V. Aute. A multi-objective genetic algorithm for robust design optimization. In *Proceedings of Genetic and Evolutionary Computation Conference 05*, pages 771–778, 2005.

[55] Y. Liang, X. Q. Cheng, Z. N. Li, and J. W. Xiang. Robust multi-objective wing design optimization via CFD approximation model. *Engineering Applications of Computational Fluid Mechanics*, 5(2):286–300, 2011.

[56] K. Miettinen. *Nonlinear Multiobjective Optimization*. Kluwer Academic Publishers, 1999.

[57] K. Miettinen. Survey of methods to visualize alternatives in multiple criteria decision making problems. *OR Spectrum*, 36(1):3–37, 2014.

[58] K. Miettinen and M. M. Mäkelä. Interactive bundle-based method for nondifferentiable multiobjective optimization: NIMBUS. *Optimization*, 34(3):231–246, 1995.

[59] K. Miettinen and M. M. Mäkelä. Interactive method NIMBUS for nondifferentiable multiobjective optimization problems. In J. Clímaco, editor, *Multicriteria Analysis*, pages 310–319. Springer, 1997.

[60] K. Miettinen and M. M. Mäkelä. Synchronous approach in interactive multiobjective optimization. *European Journal of Operational Research*, 170(3):909–922, 2006.

[61] K. Miettinen, J. Mustajoki, and T. J. Stewart. Interactive multiobjective optimization with NIMBUS for decision making under uncertainty. *OR Spectrum*, 36(1):39–56, 2014.

[62] S. Narula and H. Weistroffer. A flexible method for nonlinear multicriteria decision-making problems. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(4):883–887, 1989.

[63] Y. Nikulin, O. Karelkina, and M. M. Mäkelä. On accuracy, robustness and tolerances in vector Boolean optimization. *European Journal of Operational Research*, 224(3):449 – 457, 2013.

[64] V. Ojalehto, K. Miettinen, and T. Laukkanen. Implementation aspects of interactive multiobjective optimization for modeling environments: The case of GAMS-NIMBUS. *Computational Optimization and Applications*, 58(3):757–779, 2014.

[65] V. Ojalehto, D. Podkopaev, and K. Miettinen. Towards automatic testing of reference point based interactive methods. In J. Handl, E. Hart, P. R. Lewis, M. López-Ibáñez, G. Ochoa, and B. Paechter, editors, *Parallel Problem Solving from Nature – PPSN XIV*, pages 483–492. Springer, 2016.

[66] A. Raith, M. Schmidt, A. Schöbel, and L. Thom. Extensions of labeling algorithms for multiobjective uncertain shortest path problems. *Networks*, 2018. doi: 10.1002/net.21815.

[67] A. Raith, M. Schmidt, A. Schöbel, and L. Thom. Multi-objective minmax robust combinatorial optimization with cardinality-constrained uncertainty. *European Journal of Operational Research*, 267(2):628 – 642, 2018.

[68] C. L. Sabioni, M. F. de Oliveira Ribeiro, and J. A. de Vasconcelos. Decision maker iterative-based framework for multiobjective robust optimization. *Neurocomputing*, 242:113–130, 2017.

[69] A. Saltelli, K. Chan, and E. M. Scott. *Sensitivity Analysis*. John Wileyiley & Sons, 2000.

[70] Y. Sawaragi, H. Nakayama, and T. Tanino. *Theory of Multiobjective Optimization*. Academic Press, 1985.

[71] A. Schöbel. Generalized light robustness and the trade-off between robustness and nominal quality. *Mathematical Methods of Operations Research*, 80(2):161–191, 2014.

[72] K. Shimoyama, A. Oyama, and K. Fujii. Development of multi-objective six sigma approach for robust design optimization. *Journal of Aerospace Computing, Information, and Communication*, 5(8):215–233, 2008.

[73] A. Soyster. Convex programming with set-inclusive constraints and applications to inexact linear programming. *Operations Research*, 21(5):1154–1157, 1973.

[74] R. E. Steuer. *Multiple Criteria Optimization: Theory, Computation, and Applications*. John Wiley & Sons, Inc., 1986.

[75] R. E. Steuer and E. U. Choo. Interactive weighted Tchebycheff procedure for multiple objective programming. *Mathematical Programming*, 26(3):326–344, 1983.

[76] G. Sun, G. Li, S. Zhou, H. Li, S. Hou, and Q. Li. Crashworthiness design of vehicle by using multiobjective robust optimization. *Structural and Multidisciplinary Optimization*, 44(1):99–110, 2010.

[77] T. Tulabandhula and C. Rudin. Robust Optimization using Machine Learning for Uncertainty Sets. *ArXiv e-prints*, July 2014.

[78] M. M. Wiecek, V. Y. Blouin, G. M. Fadel, A. Engau, B. J. Hunt, and V. Singh. Multi-scenario multi-objective optimization with applications in engineering design. In V. Barichard, M. Ehrgott, X. Gandibleux, and V. T'Kindt, editors, *Multiobjective Programming and Goal Programming: Theoretical Results and Practical Applications. Lecture Notes in Economics and Mathematical Systems.*, volume 618, pages 283–298. Springer, 2009.

[79] M. M. Wiecek and G. M. Dranichak. Robust multiobjective optimization for decision making under uncertainty and conflict. In A. Gupta, A. Capponi, J. C. Smith, and H. J. Greenberg, editors, *Optimization Challenges in Complex, Networked and Risky Systems*, pages 84–114, 2016.

[80] A. P. Wierzbicki. The use of reference objectives in multiobjective optimization. In G. Fandel and T. Gal, editors, *Multiple Criteria Decision Making Theory and Application*, pages 468–486. Springer Berlin Heidelberg, 1980.

[81] A. P. Wierzbicki. On the completeness and constructiveness of parametric characterizations to vector optimization problems. *OR Spectrum*, 8(2):73–87, 1986.

[82] A. P. Wierzbicki. Reference point approaches. In T. Gal, T. J. Stewart, and T. Hanne, editors, *Multicriteria Decision Making: Advances in MCDM Models, Algorithms, Theory, and Applications. International Series in Operations Research & Management Science*, volume 21. Springer, 1999.

[83] P. Xidonas, G. Mavrotas, C. Hassapis, and C. Zopounidis. Robust multiobjective portfolio optimization: A minimax regret approach. *European Journal of Operational Research*, 262(1):299 – 305, 2017.

[84] J. B. Yang. Minimax reference point approach and its application for multiobjective optimization. *European Journal of Operational Research*, 126(3):541–556, 2000.

[85] M. Zamani, M. Soleimani-damaneh, and A. Kabgani. Robustness in nonsmooth nonlinear multi-objective programming. *European Journal of Operational Research*, 247(2):370–378, 2015.

[86] Q. Zhang and H. Li. MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on Evolutionary Computation*, 11(6):712–731, 2007.

[87] E. Zitzler, D. Brockhoff, and L. Thiele. The hypervolume indicator revisited: On the design of Pareto-compliant indicators via weighted integration. In S. Obayashi, K. Deb, C. Poloni, T. Hiroyasu, and T. Murata, editors, *Evolutionary Multi-Criterion Optimization, Proceedings*, pages 862–876. Springer, 2007.

[88] E. Zitzler and L. Thiele. Multiobjective optimization using evolutionary algorithms — a comparative case study. In A. E. Eiben, T. Bäck, M. Schoenauer, and H.-P. Schwefel, editors, *Parallel Problem Solving from Nature — PPSN V, Proceedings*, pages 292–301. Springer, 1998.

[89] E. Zitzler, L. Thiele, and J. Bader. On set-based multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 14(1):58–79, 2010.

[90] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. G. da Fonseca. Performance assessment of multiobjective optimizers: an analysis and review. *IEEE Transactions on Evolutionary Computation*, 7(2):117–132, 2003.

ORIGINAL PAPERS

PI

SOLVING MULTIOBJECTIVE OPTIMIZATION PROBLEMS WITH DECISION UNCERTAINTY: AN INTERACTIVE APPROACH

by

Yue Zhou-Kangas, Kaisa Miettinen, and Karthik Sindhya 2018

Journal of Business Economics, to appear, doi: 10.1007/s11573-018-0900-1

J Bus Econ
https://doi.org/10.1007/s11573-018-0900-1

ORIGINAL PAPER

Solving multiobjective optimization problems with decision uncertainty: an interactive approach

Yue Zhou-Kangas[1] · **Kaisa Miettinen**[1] · **Karthik Sindhya**[1]

Abstract We propose an interactive approach to support a decision maker to find a most preferred robust solution to multiobjective optimization problems with decision uncertainty. A new robustness measure that is understandable for the decision maker is incorporated as an additional objective in the problem formulation. The proposed interactive approach utilizes elements of the synchronous NIMBUS method and is aimed at supporting the decision maker to consider the objective function values and the robustness of a solution simultaneously. In the interactive approach, we offer different alternatives for the decision maker to express her/his preferences related to the robustness of a solution. To consolidate the interactive approach, we tailor a visualization to illustrate both the objective function values and the robustness of a solution. We demonstrate the advantages of the interactive approach by solving example problems.

Keywords Multiple criteria decision making · Robust solutions · Interactive methods · Handling uncertainties · NIMBUS · Robustness measure

JEL Classification C61

✉ Yue Zhou-Kangas
yue.y.zhou-kangas@jyu.fi

Kaisa Miettinen
kaisa.miettinen@jyu.fi

Karthik Sindhya
karthik.sindhya@jyu.fi

[1] University of Jyvaskyla, Faculty of Information Technology, PO Box 35 (Agora), FI-40014 University of Jyvaskyla, Finland

Published online: 15 March 2018

1 Introduction

Practical optimization problems often involve multiple conflicting objectives. For these problems, there does not exist a single optimal solution. Instead, there is a set of mathematically equally good Pareto optimal solutions. A solution is Pareto optimal, if none of the objective function values can be improved without impairing at least one of the others. Typically, a decision maker (DM) who is an expert in the problem domain is interested in a single Pareto optimal solution depending on her/his preferences (Branke et al. 2008) and can be assumed to provide this preference information.

In optimization, uncertainty can originate from different sources and be reflected in different elements of the problems such as decision variables and parameters in objectives or constraints (Wiecek and Dranichak 2016). For example, in portfolio optimization, uncertain future developments can be reflected as parameters in the objective functions as in Miettinen et al. (2014). Furthermore, uncertainty from possible changes on government policy can be reflected in constraints as in Hassanzadeh et al. (2013) and asset classes cannot be held exactly as planned due to for example change of the regulations can be reflected as uncertainty in decision variables.

In conventional multiobjective optimization methods, the involvement of uncertainty in the problems is usually ignored. As a result, the immunity to uncertainty which we call robustness of solutions is not considered. However, the uncertainty can render the optimized solutions ineffective with undesirable and unexpected degradation on the objective function values. Thus, the consideration on robustness of solutions is as relevant as that of multiple objectives for practical problems.

When robustness is considered, a multiobjective optimization method has to find Pareto optimal solutions without knowing the behavior of the uncertain data exactly. Consequently, a DM has to understand the consequences of the involved uncertainty in addition to considering multiple conflicting objectives simultaneously. In addition, the DM also needs to learn the possible trade-off between the objective function values and robustness.

In recent years, different approaches have been developed sharing the common goal of identifying solutions both with respect to multiple conflicting objectives and being sufficiently immune to the uncertainty (see e.g., Azaron et al. 2008, Ehrgott et al. 2014, and Talaei et al. 2016). However, in this paper, we do not assume the availability of probability distribution information as in Azaron et al. (2008) because such information is not always available. On the other hand, we do not expect any deep understanding on the problem from the DM to judge a fuzzy membership as in Talaei et al. (2016). Instead, we aim at supporting the DM to learn about the problem, its attainable solutions, and the consequences of uncertainty and eventually find a most preferred solution.

Different multiobjective optimization methods (see e.g., Miettinen 1999, Sawaragi et al. 1985 and Steuer 1986), can be classified into a priori, a posteriori, and interactive methods (see e.g., Miettinen 1999). Interactive methods has demonstrated advantages in supporting a DM to iteratively find a most preferred solution. With interactive methods, the DM does not need to know her/his preferences before knowing the attainable solutions as in so-called a priori methods. (S)he is not expected to make a choice among a (large) set of solutions as in a posteriori methods which can be cognitively challenging. Instead, the DM guides the solution process by specifying her/his preferences in each iteration based on a given Pareto optimal solution. During this process, the DM is provided the opportunity to learn about the problem and its attainable solutions. The possibility of learning demonstrates strong potential for us to utilize to achieve the aim. Thus, we concentrate on exploiting interactive methods to support the DM to make a well-informed decision.

As mentioned before, uncertainty can be reflected in different elements of a multiobjective optimization problem. For problems with parameter uncertainty, different so-called robust Pareto optimal solutions (see summary in Ide and Schöbel 2016 and Wiecek and Dranichak 2016) have been defined and some solution methods have been proposed in Bokrantz and Fredriksson (2017) and Ehrgott et al. (2014). There have been also attempts to support the DM with interactive methods e.g., in Hassanzadeh et al. (2013) and Miettinen et al. (2014). We concentrate on decision uncertainty (Eichfelder et al. 2017) in this paper because of the lack of research efforts in supporting the DM to find a most preferred solution for this type of problems.

By incorporation of robustness, we mean the analysis on the consequences of decision uncertainty in the objective function values. In the literature, there exist at least three different strategies to incorporate robustness in solving problems with decision uncertainty, but none of them concentrates on supporting the DM. The first type of strategy is to combine additional objectives, which quantify the robustness of a solution, with the original objectives as in Asafuddoula et al. (2015) and Gaspar-Cunha and Covas (2007). The changes in the objective function values due to the uncertainty are optimized simultaneously with the original objectives to compute a set of solutions. From these solutions, the DM is expected to select one based on her/his preferences.

Second, problems with decision uncertainty can be transformed to deterministic ones by modifying the objectives. In Eichfelder et al. (2017), the concept of regularization robustness is extended to multiobjective optimization problems to derive a regularized robust counterpart of the uncertain problem. In Deb and Gupta (2006), the original objective functions are replaced by the so-called mean effective objective functions. In Liang et al. (2011) and Sun et al. (2010), the original objective functions are replaced by their approximated mean and variance functions.

Third, a robustness measure can be used as an additional constraint as in Deb and Gupta (2006), Gunawan and Azarm (2005) and Li et al. (2005) where only solutions whose measured robustness satisfies predefined thresholds are considered feasible. Alternatively, a set of Pareto optimal or near-Pareto-optimal solutions can be compared based on their measured robustness as in Barrico and Antunes (2006) and Salimi and Lowther (2016).

Even though the first type of strategy allows the DM to consider multiple objectives and robustness simultaneously, additional objective functions can bring additional cognitive load to the DM. For example, when the deviation of the value of each objective function is combined with the original objectives as in Asafuddoula et al. (2015), the DM has to consider double amount of objectives simultaneously. Thus, the amount of additional objective functions should be minimized. To support the DM to make a well-informed decision, the information exchange in the interactive solution process should be understandable, i.e., (s)he should understand the provided information and can express her/his preferences conveniently. So the DM should be informed on the objective function values during the solution process. For this purpose, the original objective functions should be preserved. Thus, the second type of strategy is not well fitted for interactive methods. In addition, robustness measure as a constraint as in the third type of strategy does not provide the opportunity for the DM to consider it simultaneously with the objectives and (s)he cannot directly specify her/his preferences. In addition, robustness measure as such should have an understandable meaning to the DM. Thus, with the focus on supporting the DM to learn about the problem and the consequences of uncertainty, we need further developments.

Motivated by the gaps in the literature, we quantify the robustness of solutions with a single understandable robustness measure to capture the consequences of decision uncertainty in the multiple objectives in the problem. Together with the original multiple objectives, we add the robustness measure as an additional objective to give the DM the opportunity of considering robustness and objective function values simultaneously and, thus, balancing between robustness and desirable objective function values. Our goal is not to develop a totally new interactive method but to enhance the existing ones when decision uncertainty is involved in the problem. As an example, we utilize elements of synchronous NIMBUS (Miettinen and Mäkelä 2006). But our approach can also be used in for example reference point-based methods (Wierzbicki 1982).

To support the DM to learn about the consequences of the uncertainty during an interactive solution process, a robustness measure should include the following desired properties: 1. The numerical value should reflect how the uncertainties in decision variables can affect the objective function values. Based on the value, the DM can consider how 'robust' a solution is. 2. With the computed numerical value, the DM should be able to specify her/his preferences conveniently. Based on these desired properties, we first identify and analyze the robustness measures in the literature that are closest to them. Then we propose an alternative robustness measure which meets both of the desired properties, and can thus better support the DM in an interactive approach.

The rest of the paper is organized as follows: in Sect. 2, we present some basic concepts, introduce the NIMBUS method briefly, and discuss robustness measures that are closest to the desired properties. Then in Sect. 3, we present a robustness measure that can be understandable for the DM and propose the interactive approach, which is followed by numerical examples where we demonstrate the advantages of our approach by solving two problems in Sect. 4. Finally, we conclude the paper in Sect. 5.

2 Background

2.1 Multiobjective optimization and decision uncertainty

Deterministic multiobjective optimization problems are defined in the form

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

with objective functions (objectives) $f_i : S \rightarrow \mathbb{R}$ to be simultaneously optimized, where $1 \leq i \leq k$ and $k \geq 2$. The decision vectors (which consist of decision variables as their components) $\boldsymbol{x} = (x_1, \ldots, x_n)^T$ belong to the nonempty feasible region $S \subset \mathbb{R}^n$. Objective vectors $\boldsymbol{f}(\boldsymbol{x}) = (f_1(\boldsymbol{x}), \ldots, f_k(\boldsymbol{x}))^T$ consist of objective function values which are the images of the decision vectors. The image of the feasible region is called the feasible objective region $Z = \boldsymbol{f}(S)$. If all the objective functions are minimized, a solution $\bar{\boldsymbol{x}}$ is said to be Pareto optimal if there does not exist another solution $\boldsymbol{x} \in S$ such that $f_i(\boldsymbol{x}) \leq f_i(\bar{\boldsymbol{x}})$ for all $i = 1, \ldots, k$ and the inequality is strict for at least one index j. If some of the objectives f_i is to be maximized, it is equivalent to minimize $-f_i$.

For (1), the set of Pareto optimal solutions usually contains more than one elements. Mathematically, Pareto optimal solutions are incomparable. The DM is expected to identify the most preferred one among them as the final solution. Only one DM is assumed to be involved in the solution process in this paper. It is often useful for the DM to know the ranges of the objective function values in the set of Pareto optimal solutions. The ideal objective vector $\boldsymbol{z}^* = (z_1^*, \ldots, z_k^*)^T$ and the nadir objective vector $\boldsymbol{z}^{nad} = (z_1^{nad}, \ldots, z_k^{nad})^T$ give the bounds of the objective function values. The ideal objective vector is formed by the individual optima of each objective function in the feasible region. The utopian objective vector $\boldsymbol{z}^{**}$, which is strictly better than z^*, is defined for computational reasons. In practice, z_i^{**} is set as $z_i^* - \epsilon$ for $i = 1, \ldots, k$ if f_i is to be minimized, where $\epsilon > 0$ 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 e.g., Miettinen 1999 for further details). If the objective function values are with different magnitudes, the nadir and utopian objective vectors can be used to normalize them.

In this paper, we consider multiobjective optimization problems with decision uncertainty. By decision uncertainty, we mean that a computed solution, which we refer to as the base solution $\boldsymbol{x}^b$, cannot be guaranteed to be implemented exactly. Instead, the implementation can involve small perturbations $\boldsymbol{\Delta x}$, i.e., the implemented solution is from the set $\{\boldsymbol{x}^b + \boldsymbol{\Delta x} | \boldsymbol{\Delta x} \in \Omega\}$ where Ω is the set of all possible perturbations in the neighborhood of the base solution. We assume that Ω is a hyperbox and $\boldsymbol{0} \in \Omega$, which does not have to be sysmetric. We refer to the corresponding objective vector $\boldsymbol{f}(\boldsymbol{x}^b)$ as the base objective vector whose components are the base objective function values. The type of uncertain multiobjective optimization problems considered is of the form:

$$\begin{array}{ll} \text{minimize or maximize} & \{f_1(\boldsymbol{x} + \boldsymbol{\Delta x}), \ldots, f_k(\boldsymbol{x} + \boldsymbol{\Delta x})\} \\ \text{subject to} & \boldsymbol{x} \in S \\ & \boldsymbol{x} + \boldsymbol{\Delta x} \in S, \text{for all } \boldsymbol{\Delta x} \in \Omega. \end{array} \tag{2}$$

In the formulation, $\boldsymbol{x}$ is the decision vector and $\boldsymbol{\Delta x}$ is the unknown possible perturbation within the hyperbox Ω. To solve this problem, we consider all the possible values $\boldsymbol{\Delta x} \in \Omega$ and search for a most satisfactory base solution $\boldsymbol{x}^b$ for the DM. By a most satisfactory base solution, we mean that the DM is satisfied with the base objective function values $(f_1(\boldsymbol{x}^b), \dots, f_k(\boldsymbol{x}^b))^T$ and the objective function values when perturbations occur, i.e., $(f_1(\boldsymbol{x}^b + \boldsymbol{\Delta x}), \dots, f_k(\boldsymbol{x}^b + \boldsymbol{\Delta x}))^T$ for all $\boldsymbol{\Delta x} \in \Omega$.

2.2 NIMBUS

As mentioned in Sect. 1, we utilize elements of the NIMBUS method to build our interactive approach. As mentioned in Miettinen and Mäkelä (2006), one can always derive a reference point from the preference information utilized in NIMBUS, and thus, our approach to be proposed can also be used with reference point based methods. In NIMBUS, given the current Pareto optimal solution $\boldsymbol{x}^c$, the DM directs the interactive solution process by specifying preferences as a classification of the objectives. The classification indicates how the current objective function values $\boldsymbol{f}(\boldsymbol{x}^c)$ should change to be more desired by the DM. The DM can classify the objective functions into 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 until a bound ϵ_i, and
$I^\diamond$ for those that are temporarily allowed to change freely.

Each objective is assigned to one of the classes described above. Some objectives must be allowed to be impaired to enable improvements in others because of the nature of the Pareto optimality. If aspiration levels or bounds are used, the DM is expected to provide them.

In the NIMBUS method, new Pareto optimal solutions are computed by solving a scalarized problem, which includes preference information given by the DM in the classifications. In this paper, we use one of the four scalarized problems of the synchronous NIMBUS method (Miettinen and Mäkelä 2006), which has the form (for minimizing the objectives):

$$\begin{aligned}
\text{minimize} \quad & \max_{\substack{i \in I^< \\ j \in I^\leq}} \{w_i(f_i(\boldsymbol{x}) - z_i^*), w_j(f_j(\boldsymbol{x}) - \hat{z}_j)\} + \rho \sum_{i=1}^{k} w_i f_i(\boldsymbol{x}) \\
\text{subject to} \quad & \boldsymbol{x} \in S \\
& f_i(\boldsymbol{x}) \leq f_i(\boldsymbol{x}^c) \text{ for all } i \in I^< \cup I^\leq \cup I^=, \\
& f_i(\boldsymbol{x}) \leq \epsilon_i \text{ for all } i \in I^\geq,
\end{aligned} \tag{3}$$

where $\boldsymbol{x}^c$ is the current Pareto optimal solution, $\boldsymbol{z}^*$ is the ideal objective vector, $\hat{z}_i$ are the aspiration levels for the objective functions in $I^{\leq}$, ϵ_i are the bounds of allowed impairment for the objective functions in $I^{\geq}$, $\rho > 0$ is a small scalar bounding the trade-offs, and the coefficients $w_i\,(1 \leq i \leq k)$ are constants used for scaling the objectives. The value of w_i is based on the estimated ranges, i.e., $\frac{1}{z_i^{nad} - z_i^{**}}$, for normalizing the objective function values.

The DM can compare the two Pareto optimal solutions before and after the classification, so that (s)he can learn how attainable her/his desired changes were. For more information about the method and the proof of Pareto optimality, see Miettinen and Mäkelä (2006). In addition, the NIMBUS method provides the DM an opportunity to generate intermediate solutions and to save interesting solutions during the iterative solution process. The DM can return to a saved solution any time or select one as the most preferred solution from the set of saved solutions.

We shall return to the NIMBUS method and discuss our interactive approach for solving multiobjective optimization problems with decision uncertainty in Sect. 3. In what follows, we discuss the robustness measures from the literature.

2.3 Robustness measures from the literature

As discussed in Sect. 1, objective functions are assumed to have a meaning to the DM and, thus, the original objective functions should be preserved to allow the DM to consider their values together with the robustness simultaneously. Furthermore, the DM should also be able to learn their possible trade-offs. For this purpose, naturally, a robustness measure should be used as an additional objective to the problem formulation, i.e., we solve a multiobjective optimization problem by combining the original objectives and a robustness measure as its objectives. By employing an additional objective, the DM can consider balancing between robustness and objective function values.

We have identified three robustness measures in the literature that are closest to the desired properties to be used in the interactive solution process as listed in the introduction. These measures quantify the robustness of a base solution and were originally used as additional constraints, but they can be used as an additional objective as well.

In Deb and Gupta (2006), the measure involves sampling and the difference between the base objective vector and the average function values of samples in the neighborhood of a base solution is used to measure its robustness:

$$R_1(\boldsymbol{x}^b) = \frac{\|\boldsymbol{f}^p(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{x}^b)\|}{\|\boldsymbol{f}(\boldsymbol{x}^b)\|}, \tag{4}$$

where $\|\cdot\|$ is the Euclidean norm. The so-called mean effective objective vector $\boldsymbol{f}^p(\boldsymbol{x})$ consists of the average objective function values of the samples in the neighborhood. According to the definition, the smaller the value of $R_1(\boldsymbol{x}^b)$ is, the more robust the solution is. We can adopt this measure as an additional objective to be minimized.

In Gaspar-Cunha and Covas (2007), a measure, which is also based on studying samples in the neighborhood of a base solution, is defined for each objective function to capture robustness:

$$f_i^{R_2}(\boldsymbol{x}^b) = \frac{1}{H} \sum_{h=1}^{H} \frac{|\tilde{f}_i(\boldsymbol{x}^h) - \tilde{f}_i(\boldsymbol{x}^b)|}{||\boldsymbol{x}^h - \boldsymbol{x}^b||}, \tag{5}$$

where $\boldsymbol{x}^h$ is a sample and H is the total number of samples. The notation $\tilde{f}_i$ is used to indicate that the objective function values are normalized within their ideal and nadir values. This measure studies how much the objective function value changes relative to the perturbation of a base solution. Based on the robustness measure of each objective function, so-called global robustness measures which include all k objectives are defined as $R_2(\boldsymbol{x}^b) = \max_{i=1,\ldots,k} f_i^{R_2}(\boldsymbol{x}^b)$, and $R_2(\boldsymbol{x}^b) = \frac{1}{k} \sum_{i=1}^{k} f_i^{R_2}(\boldsymbol{x}^b)$. We can adopt either as an additional objective to be minimized.

In Gunawan and Azarm (2005) and Li et al. (2005), given a base solution and the maximum acceptable changes from the base objective function values, the radius of the smallest hyper-sphere centered on the base solution is calculated to measure its robustness by solving a single-objective optimization problem:

$$\begin{array}{ll} \text{minimize} & ||\Delta x||_p \\ \text{subject to} & \max(\frac{|\Delta f_i|}{\Delta f_{0,i}}) = 1. \end{array} \tag{6}$$

This measure studies how much perturbation, i.e., the optimal value of Δx, is allowed in the base solution for the objective function values to be acceptable. The constraint, where Δf_i is a function of Δx, states that the maximum change in the objective function values has to be equal to the pre-specified acceptable level $\Delta f_{0,i}$. The optimized objective function value of (6) is the value of the robustness measure, which we refer to as $R_3(\boldsymbol{x}^b)$. A bigger value $R_3(\boldsymbol{x}^b)$ means that the more robust the base solution is. We can adopt this measure as an additional objective to be maximized.

We summarize the characteristics of the three measures in Table 1. The required parameters to compute the robustness measures are presented in the first

Table 1 Summary of different robustness measures

Measures	$R_1(\boldsymbol{x}^b)$	$R_2(\boldsymbol{x}^b)$	$R_3(\boldsymbol{x}^b)$
Parameters	Ω, H	Ω, H	Lower and upper bounds of Δx, value of $\Delta f_{0,i}$
Randomness involved	Random	Random	Exact and stable

row. The randomness involved in the computed values of the robustness measures is presented in the second row.

The three measures are based on the study of the neighborhood of a base solution and they require some parameters. As shown in the table, the size of the neighborhood, which is represented by Ω in (2), is commonly required. For measures $R_1(\boldsymbol{x}^b)$ and $R_2(\boldsymbol{x}^b)$, the number of samples (H) in the neighborhood is needed. There were no clear guidelines how these parameters should be set in the papers where the measures were originally proposed. For $R_3(\boldsymbol{x}^b)$, the acceptable levels of change from base objective function values $\Delta f_{0,i}$ are required. This parameter is said to be set by the user, which can be understood as the DM in our context.

Unfortunately, all the existing robustness measures have some shortcomings. As can be seen in the definitions of the measures, the numerical values do not have a direct meaning on how robust a solution is for the DM except the intuitive indication based on if the measure should be minimized or maximized. With the numerical values, the DM cannot formulate and specify her/his preferences conveniently during the interactive solution process. In the interactive solution process, the DM can only specify her/his preferences on the measures based on this intuition which does not help the DM to formulate her/his preferences clearly. The purpose of using an additional objective to incorporate the robustness is to support the DM to find a most preferred solution by simultaneously considering the base objective function values and the robustness of a solution. We can summarize that none of the robustness measures meets our needs well. To communicate the meaning of robustness to the DM in a more understandable way and to allow the DM to formulate and specify preferences conveniently, it is desirable to formulate a new robustness measure. We propose such a measure in the next section.

3 Interactive approach for solving problems with decision uncertainty

3.1 A new robustness measure

As discussed before, we incorporate a robustness measure to the problem formulation by adding an additional objective. None of the measures identified from the literature fully meets the desired properties to be incorporated in an interactive solution process. In this section, we first describe a new robustness measure which is suitable to be used in an interactive approach. Then we propose the interactive approach tailored to solve multiobjective optimization problems with decision uncertainty.

We propose a robustness measure which can deliver the meaning of robustness to the DM in a more understandable way. Our robustness measure investigates the ranges of objective function values in the neighborhood Ω of a base solution $\boldsymbol{x}^b$. The existence of the ranges of the objective function values is a consequence of the uncertainty in the decision variables. In the form of the ranges, the DM can get the information on how the objective function values change. In other words, the ranges characterized by best and worst objective function values describe the variations due to the possible perturbations in the decision variables. For an objective function f_i, the range of its value in the neighborhood can be defined as

$r_i(\boldsymbol{x}^b) = \max_{\Delta \boldsymbol{x} \in \Omega} f_i(\boldsymbol{x}^b + \Delta \boldsymbol{x}) - \min_{\Delta \boldsymbol{x} \in \Omega} f_i(\boldsymbol{x}^b + \Delta \boldsymbol{x})$. We refer to these ranges as r_i ranges. As discussed before, we want to introduce only one additional objective not to introduce too much cognitive load. We can use the maximum range, i.e., the upper bound, of all objective functions to measure the robustness of a solution as an objective to be minimized. So our robustness measure is:

$$R_4(\boldsymbol{x}^b) = \max_i \left[\frac{r_i(\boldsymbol{x}^b)}{z_i^{nad} - z_i^{**}} \right], i = 1, \dots, k, \tag{7}$$

where the lower and upper bounds of the neighborhood Ω are provided by the DM when the problem is formulated. As an expert in the application domain, the DM is more likely to know reasonable bounds than others. When compared to the robustness measures discussed in Sect. 2, Ω has the same meaning as in the measures $R_1(\boldsymbol{x}^b)$ and $R_2(\boldsymbol{x}^b)$. In this measure, the lower and upper bounds do not have to be symmetric around the base solution.

As can be seen in (7), we need to solve $2k$ additional single-objective optimization problems to compute the value. In principle, this can provide the DM exact measurements of the r_i ranges. If approximated r_i ranges can be accepted by the DM, similar sampling techniques as presented in Deb and Gupta (2006) and Gaspar-Cunha and Covas (2007) for measures $R_1(\boldsymbol{x}^b)$ and $R_2(\boldsymbol{x}^b)$ can be applied. For the rest of the paper, we refer to f_i as an active objective function if i gives the maximum for $R_4(\boldsymbol{x}^b)$ in (7).

The computed value of $R_4(\boldsymbol{x}^b)$ is the percentage of the r_i range with respect to the ideal and nadir values of the active objective function. With the help of the ideal and nadir values, this numerical value can tell the DM how much the objective function varies in its own range, which is a concrete expression on the consequences of the uncertainty. The DM can also learn without much effort that the r_i ranges of the other objective functions are smaller than this value. In addition, by computing the value of $R_4(\boldsymbol{x}^b)$, the r_i ranges of all the objective function values in the neighborhood are also available, which we will utilize to support the DM in the interactive solution process. We will discuss how we can utilize the r_i ranges and develop an appropriate visualization to improve the understandability of the robustness measure in the next subsection where we discuss the proposed interactive approach.

3.2 An interactive approach for solving multiobjective optimization problems with decision uncertainty

We incorporate robustness into the problem formulation (1) by adding the measure $R_4(\boldsymbol{x}^b)$ as an additional objective. In Wiecek et al. (2009), Pareto optimality to the original problem of a Pareto optimal solution to a problem formulated with an additional objective is summarized. A solution remains Pareto optimal or not depending on how the objectives conflict with one another in the new problem with an additional objective. So whether the robust solutions found by our approach are Pareto optimal to the original problem depends on the problem itself and the consequences of uncertainty. However, as argued in Bertsimas and Sim

(2004), the robustness of a solution and the corresponding values of the original objectives usually conflict with each other. To gain robustness, sacrifices on the objective function values can be necessary. Furthermore, by learning the trade-offs between the objective function values and robustness, it is a conscious choice for the DM if objective function values are sacrificed.

As mentioned before, we build our interactive approach by utilizing the elements of the synchronous NIMBUS method. We follow the interactive solution process of the NIMBUS method where the DM is expected to specify her/his preferences by classifying the objective functions as described in Sect. 2.2. Our goal is to support the DM to find a base solution with most satisfactory base objective function values and objective function values when the perturbations occur. This involves differences from the original NIMBUS method, in which the goal is to support the DM to find a most preferred Pareto optimal solution. Because of the specific robustness consideration, we tailor some components of the NIMBUS solution process to support the DM to consider the base objective function values and the robustness of a solution simultaneously.

The numerical value of the robustness measure $R_4(\boldsymbol{x}^b)$ is used to capture the robustness of a base solution. It is the percentage of the r_i range of the active objective function for a base solution in its given neighborhood within the range of that objective. With the information of ideal and nadir values, the DM can combine the numerical value of $R_4(\boldsymbol{x}^b)$ and the ranges of the active objective function to have a concrete understanding on the robustness of the solution.

Based on the definition of $R_4(\boldsymbol{x}^b)$, the r_i ranges (in percentage) of other objective functions are guaranteed to be smaller. This allows the DM to indirectly specify her/his preferences on the r_i range of a specific objective function by providing the desired value for $R_4(\boldsymbol{x}^b)$. By doing so, the DM has specified the desired maximum r_i ranges for all the original objective functions, in which the specific objective function is included. The DM is more likely to learn about these facts without much cognitive effort than learning a numerical value without a direct meaning on the consequences of decision uncertainty. In addition, since the r_i range of each objective function is naturally available with the computation of $R_4(\boldsymbol{x}^b)$, we will utilize this information when we present a solution to the DM.

Visualization can be used to support the DM in studying the trade-offs between optimality and robustness. To visually present a solution to the DM, we tailor a visualization method for presenting the base objective function values and the robustness information simultaneously. An additional component is added to one of the visualizations used in the IND-NIMBUS framework (see Miettinen 2006 and Ojalehto et al. 2014). An example of the visualization in IND-NIMBUS is shown in Fig. 1 with two objective functions to be minimized. Each objective function is visualized as a horizontal bar within the range of its ideal and nadir objective values. The colored part of the bar illustrates the current objective function value z_i^c, which starts from the ideal value towards the nadir objective value. The DM can classify an objective function e.g., by sliding the endpoint of the colored bar. Instead of adding an additional bar for the value of $R_4(\boldsymbol{x}^b)$, we superimpose the r_i ranges on top of the corresponding bars of the k (original) objective functions.

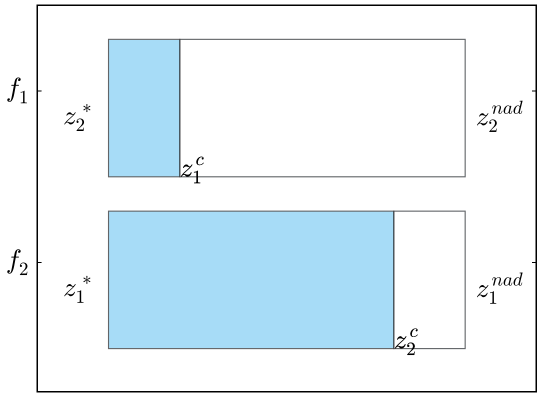


Fig. 1 Original IND-NIMBUS visualization

An example of the tailored visualization method is presented in Fig. 2. The r_i range of each objective function is presented as a gray shadow around the current base objective function value. The r_i range indicating the robustness of the current active objective function is highlighted with a frame to inform the DM that (s)he should pay attention to it and can specify her/his preferences. When we visualize a solution as a part of a solution process, the numerical values of the upper and lower bounds of variation in each objective function value will be shown in the corresponding places of the gray shadow. The value of the objective f_{k+1} will be shown in the upper right corner of the highlighted frame. This visualization can further help the DM to understand the robustness of the base solution concretely together with the value of $R_4(\boldsymbol{x}^b)$ because it can illustrate how the uncertainties in the decision variable are reflected in each objective function.

With a solution presented in terms of the base objective function values and the r_i ranges as in Fig. 2, the DM can specify her/his preferences for a more desired solution. In the original NIMBUS method, the DM is expected to classify all the objective functions and provide the aspiration levels and bounds for the corresponding

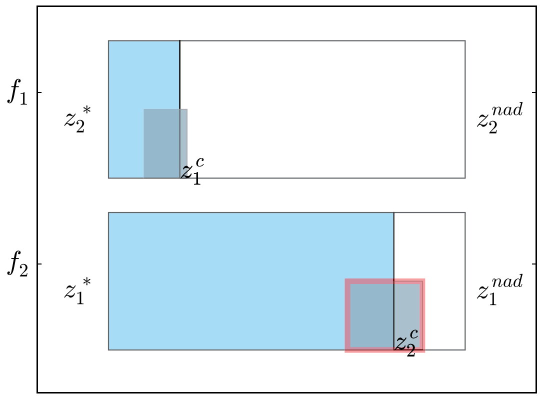


Fig. 2 Visualization with robustness information

classes of objective functions. In our approach, the DM can choose to follow the original NIMBUS method to classify the objective function for robustness in the same way as for the original objective functions. Alternatively, the DM can also choose to classify the original objective functions but specify lower and upper bounds of the r_i range on the current active objective function. In this case, we say that the DM chooses to adjust the r_i range. Once the adjusted r_i range has been specified by the DM, we convert it as a NIMBUS classification by calculating the desired aspiration level. We return to this technical detail on converting an adjusted r_i range to a proper classification later.

With the tailored visualization and the multiple available ways of expressing the preferences on the robustness of a solution, our interactive approach, as shown in the flowchart in Fig. 3, starts from computing an initial solution and presenting it to the DM with the tailored visualization. The multiple ways to specify preferences on the robustness measure are available for the DM in the highlighted intermediate step. Based on the specified preferences by the DM, we convert the preferences as classifications if necessary, then compute a new solution by solving the scalarized problem (3) and present it to the DM. If the DM is not satisfied with the solution, the solution process continues as in the original NIMBUS method. In this way, the DM iteratively guides the solution process towards a most preferred robust solution.

This interactive approach has four advantages which aim at providing better support to the DM in the solution process. First, the meaning of the numerical value of the robustness measure $R_4(\boldsymbol{x}^b)$ is understandable for the DM, because it is the percentage of maximum possible change in the objective function values with respect to the ideal and nadir values. Second, the r_i ranges presented to the DM provide an opportunity for the DM to observe and understand how the uncertainties in the

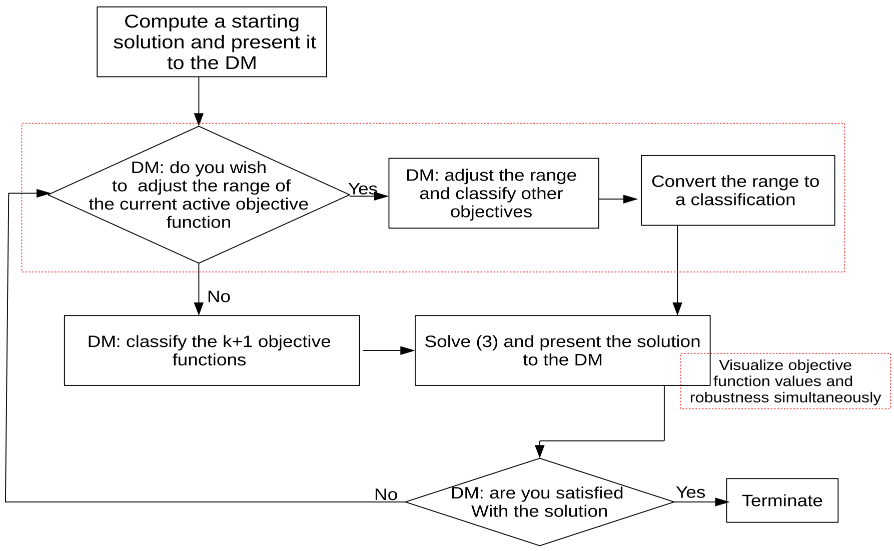


Fig. 3 Flowchart of the interactive approach

decision variables affect the objective function values, i.e., the consequences of uncertainty. In addition, the r_i ranges are presented together with the base objective function values. So the DM can consider both of them at the same time. Third, we provide multiple alternatives for the DM to specify her/his preferences concerning the robustness of a more desired solution. So in the solution process, the DM can choose a comfortable way in each iteration of the solution process. Fourth, with the robustness measure incorporated as an addition objective function, the DM can find an acceptable balance between the robustness and the objective function values of a solution and gain insights on how they are interdependent.

As mentioned before, the DM can classify all the objectives. If (s)he does so, we can proceed directly by solving (3) for a new solution. Alternatively, the DM can pay special attention to the active objective function and adjust the r_i ranges. In this case, we need to convert the adjusted r_i range to a NIMBUS classification. The close relationship between the desirable aspiration level of an objective function and the classification of it was discussed in Miettinen and Mäkelä (2002) and Miettinen and Mäkelä (2006). Here we have $f_{k+1} = R_4(\boldsymbol{x}^b)$. Depending on the adjusted r_i range, we have five different types of conversion:

- The adjusted r_i range is smaller than the current one: it means that the DM wishes to have a more robust solution and f_{k+1} is classified as to be improved to an aspiration level, i.e., $f_{k+1} \in I^{\leq}$ with a value $\hat{z}_{k+1}$, where $\hat{z}_{k+1}$ is the calculated aspiration level based on the adjusted r_i range ;
- The adjusted r_i range is greater than the current one: it means that the DM can accept a less robust solution and f_{k+1} is classified as to be impaired until an upper bound, i.e., $f_{k+1} \in I^{\geq}$ with a value ϵ_{k+1}, where ϵ_{k+1} is the calculated bound based on the adjusted r_i range in the same way as for the aspiration level;
- The adjusted r_i range is the same as the current one: it means that the DM wishes to have a solution as robust as the current one and f_{k+1} is classified as $f_{k+1} \in I^{=}$;
- The DM has adjusted the r_i range to be 0: it means that the DM wishes to have a solution as robust as possible and f_{k+1} is classified as $f_{k+1} \in I^{<}$.
- The DM does not specify any adjustments for the r_i range. This means that (s)he can accept any value in f_{k+1}. So $f_{k+1} \in I^{\diamond}$.

After the conversion, we can use the resulting classification to compute a new solution for the DM.

4 Numerical results

4.1 River pollution problem

In this section, we illustrate the solution process of two multiobjective optimization problems with decision uncertainty with the proposed interactive approach. The river pollution problem considered was originally presented in Narula and Weistroffer (1989) as a deterministic problem. In the problem, a fishery and a city are polluting water in a river. The city is located downstream from the fishery. Both the city and the fishery have their own pollution treatment plants. We consider the following formulation:

$$
\begin{array}{ll}
\text{maximize} & f_1(\boldsymbol{x}+\boldsymbol{\Delta x}) = 4.07 + 2.27(x_1 + \Delta x_1) \\
\text{maximize} & f_2(\boldsymbol{x}+\boldsymbol{\Delta x}) = 2.60 + 0.03(x_1 + \Delta x_1) + 0.02(x_2 + \Delta x_2) + \frac{0.01}{1.39-(x_1+\Delta x_1)^2} + \frac{0.30}{1.39-(x_2+\Delta x_2)^2} \\
\text{maximize} & f_3(\boldsymbol{x}+\boldsymbol{\Delta x}) = 8.21 - \frac{0.71}{1.09-(x_1+\Delta x_1)^2} \\
\text{minimize} & f_4(\boldsymbol{x}+\boldsymbol{\Delta x}) = -0.96 + \frac{0.96}{1.09-(x_2+\Delta x_2)^2} \\
\text{minimize} & f_5(\boldsymbol{x}+\boldsymbol{\Delta x}) = R_4(\boldsymbol{x}) \\
\text{subject to} & 0.3 \le x_1, x_2 \le 1.0, \\
 & \text{for all } \Delta x_1 \in [x_1 - 0.1, x_1 + 0.1] \text{ and } \Delta x_2 \in [x_2 - 0.1, x_2 + 0.1],
\end{array}
\tag{8}
$$

where there are four original objectives and the fifth objective function represents robustness. The decision variables x_1 and x_2 represent the proportional amount of biochemical oxygen demanding material to be removed from water in the treatment plants after the fishery and the city, respectively. The more biochemical oxygen demanding material is removed, the more the quality of water will improve. The unknown possible perturbations are represented by Δx_1 and Δx_2 . The information on the neighborhood near the base solution, i.e., lower and upper bounds of the perturbations, is provided by the DM. The first and second objective functions describe the quality of water after the fishery and after the city, respectively, and the third objective function describes the percentage of return on investment at the fishery. The fourth objective represents the addition of tax rate in the city. The fifth objective is the robustness measure presented in (7). We consider uncertainties originating from the operations of the pollution treatment plants. As a result, the amount of removed biochemical oxygen demanding material can involve perturbations from the base values. Consequently, the objective function values can be different from their base values. We solve this problem with the proposed interactive approach interacting with a DM.

The individual optima of the original objectives were calculated to form the ideal objective vector $z^* = (6.34, 3.45, 7.50, 0)^T$. The nadir objective vector was approximated as $z^{nad} = (4.75, 2.85, 0.32, 9.70)^T$. To get started, we set the ideal value of the robustness measure as 0, which means the perturbations of the base solution do not affect the objective function value at all. We set the nadir value as 1, which indicates that the r_i range of the active objective function in the neighborhood is as large as the range between the ideal and nadir values.

Initialization we first introduced our robustness measure to the DM in terms of what the value of f_5 means and what the r_i ranges in the visualization mean. Then we computed and presented an initial solution $z^0 = (5.95, 3.08, 6.46, 1.40, 0.36)^T$ with the tailored visualization method to the DM as shown in Fig. 4. The initial solution was calculated as in the NIMBUS method.

In Fig. 4, the bars present the water quality after the fishery WQ(F), water quality after the city WQ(C), return on investment of the fishery (ROI), and the additional tax rate in the city (TA). The colored part of a bar illustrates the current value of the corresponding objective (also given numerically) accompanied by the ideal and nadir values at its endpoints. The r_i range, where the values of its lower and upper

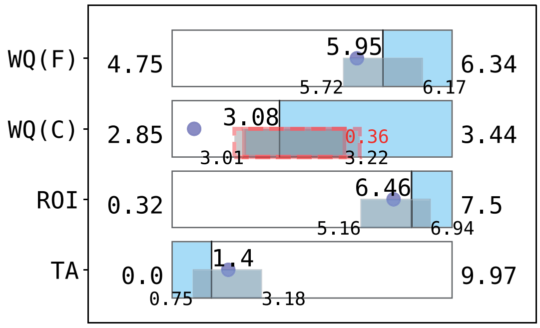


Fig. 4 Intial solution

bounds are presented at the endpoints, is superimposed on top of the corresponding bar to present how uncertainties in the solution can affect the objective function value. For the current active objective function, we highlight its r_i range and mark the current value of the robustness measure in red on the upper right corner. As discussed before, the objective function value of f_5 was not presented in its own bar. Instead, we presented the r_i ranges within the bars of the corresponding objective functions. The DM was asked to choose a preferred way to express her preferences.

The DM chose to adjust the r_i range of the current active objective function f_2. With the adjusted r_i range, we calculated an aspiration level $\epsilon_5 = 0.45$ and converted it to a classification as allowing the value of f_5 to be impaired till 0.45. The adjusted r_i range is represented as a broken line in the first illustration of Fig. 4. When considering the initial solution z^0, the DM wanted to improve the quality of water after the fishery slightly and reduce the additional tax rate to 2% in the city. At the same time, the quality of water after the city was allowed to be impaired till 2.9 and the return on investment of the fishery was also allowed to reduce till 6%. In the NIMBUS notation, the DM provided the classification for iteration 1: $I^{\leq} = \{f_1, f_4\}$ with aspiration level $\hat{z}_1 = 5.8$, and $\hat{z}_4 = 2$; $I^{\geq} = \{f_2, f_3, f_5\}$ with the bounds $\epsilon_2 = 2.9$, $\epsilon_3 = 6$, and $\epsilon_5 = 0.45$. The aspiration levels and bounds are denoted by dots in Fig. 4.

Iteration 1 based on the classification, a new solution $z^1 = (6.04, 2.94, 6.12, 0.31, 0.40)^T$ was calculated by solving the scalarized problem (3) and presented to the DM as in Fig. 5. Based on z^1, the DM could see that her preferences in iteration 1 were satisfied. However, she thought that better quality of water after the city should be achieved at the same time of maintaining the same quality of water after the fishery at the current level. In addition, she wished to maintain the current value of the maximum r_i range, i.e., no change to the robustness. Keeping in mind that some objectives have to be impaired in order to achieve better quality of water after the city, the DM allowed the return on investment of the fishery to reduce to 6% and the additional tax rate in the city till 1%. In other words, the DM gave her preference as: $I^{\geq} = \{f_3, f_4\}$ with bounds $\epsilon_3 = 6$, and $\epsilon_4 = 1$; $I^{=} = \{f_1, f_5\}$; $I^{\leq} = \{f_2\}$ with $\hat{z_2} = 3.1$.

Iteration 2 according to this classification, a new solution $z^2 = (6.04, 3.03, 6.12, 0.98, 0.39)^T$ was calculated and presented to the DM as in Fig. 6. The DM was not satisfied and wanted to make a new classification. Based

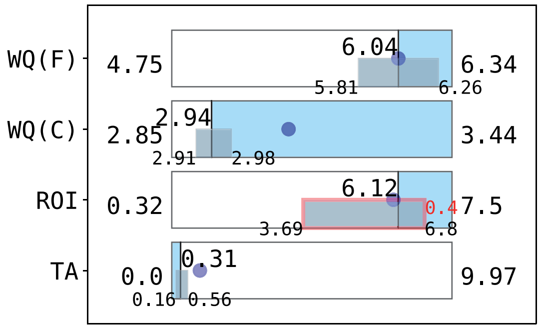


Fig. 5 Iteration 1

on z^2, the DM noticed that the additional tax rate in the city almost approached her specified upper bound and she did not wish to have worse quality of water after the city. So she decided to keep the current quality of water after the city. As an exploration of a more robust solution, she wanted to reduce the percentage of the r_i range of the active objective function to 0.3 and allow the quality of water after the fishery to decrease to 5.8, return on investment of the fishery to 5.5% and the additional tax rate in the city to increase till 2%. The DM's classification in NIMBUS notation was: $I^{=} = \{f_2\}$; $I^{\leq} = \{f_5\}$ with aspiration level $\hat{z}_5 = 0.3$; $I^{\geq} = \{f_1, f_3, f_4\}$ with bounds $\epsilon_1 = 5.8$, $\epsilon_3 = 5.5$, and $\epsilon_4 = 2$.

Iteration 3 based on the classification, the new solution computed was $z^3 = (5.86, 3.04, 6.7, 1.04, 0.29)^T$ as visualized in Fig. 7. The DM noticed that her preferences were not fully satisfied. So she tried with another classification, i.e., she did not want to have worst quality of water after the city and wanted to reduce the r_i range of the active objective function to 0.25. She continued by allowing the water quality after the fishery to be impaired till 5.8 and the return on investment of the fishery till 5.5%. The DM's classification was: $I^{=} = f_4$; $I^{\leq} = \{f_2, f_5\}$ with $\hat{z}_2 = 3.15$, and $\hat{z}_5 = 0.25$; $I^{\geq} = \{f_1, f_3\}$ with bounds $\epsilon_1 = 5.8$, and $\epsilon_3 = 5.5$.

Iteration 4 based on this classification, $z^4 = (5.86, 3.04, 6.70, 1.04, 0.29)^T$ was computed and visualized as in Fig. 8. Based on z^4, the DM noticed that the quality of water after the city did not improve as she wanted and the r_i range of the active function value did not reduce as she wished either. So she decided to accept the

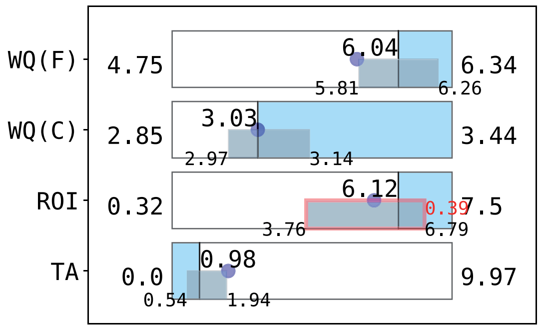


Fig. 6 Iteration 2

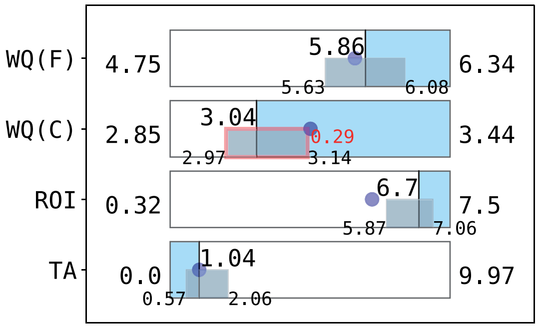


Fig. 7 Iteration 3

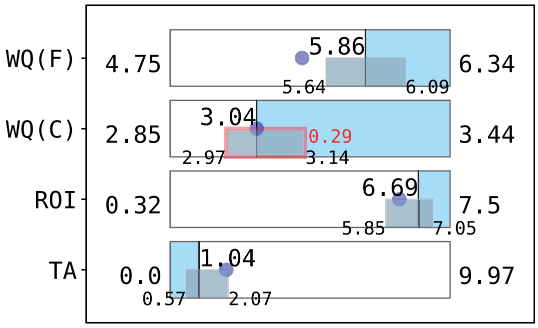


Fig. 8 Iteration 4

values of these two objectives. In addition, the return on investment of the fishery approached the bound, but the quality of water after the fishery was much better than the value she could accept. So she thought she could have a better return on investment by having a worse but acceptable quality of water after the fishery. In order to guarantee the operation of the water treatment plant in the city, thus maintaining the quality of water after the city, she decided to allow up to 2% of additional tax in the city. In NIMBUS notation, the DM specified the classification to compute z^5 as: $I^{=} = \{f_2, f_5\}$; $I^{\leq} = \{f_3\}$ with an aspiration level $\hat{z_3} = 6.2$; $I^{\geq} = \{f_1, f_4\}$ with bounds $\epsilon_1 = 5.5$, and $\epsilon_4 = 2$.

Termination with the classification, $z^5 = (5.79, 3.04, 6.84, 1.06, 0.28)^T$ was computed and presented to the DM as in Fig. 9. The DM noticed that the values of return on investment and the additional tax rate were better than she expected, but the quality of water after the fishery was already on the worst acceptable value. At the same time, the quality of water after the city and the r_i range of the active objective function were maintained. So after this iteration, the DM decided to terminate the solution process and accept z^5 as the final solution.

During the solution process, the DM was able to understand the consequences of the uncertainty via the robustness measure. Thus the final accepted solution offered a well-informed balance between the base objective function values and the robustness. At the beginning of the solution process, the DM chose to adjust the r_i range of the active objective function because she felt it would be easier. In the later

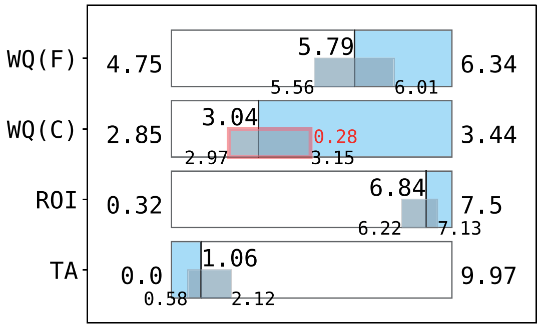


Fig. 9 Final solution

iteration, she could directly classify f_5. The DM noticed that the active objective function changed. She assumed that by altering the r_i range of the water quality after the city, the r_i range of that particular objective function would become worse. But she soon learned that the meaning of the robustness measure specified value for the active objective function is actually an upper bound for the r_i ranges of all the four objectives.

As for the advantages mentioned in Sect. 3, the DM was able to understand the meaning of the numerical value of the robustness measure. The simultaneously illustrated r_i ranges and the base objective function values helped her to consider both types of information together and then formulate and specify her preferences. She can also easily specify preferences on the robustness of a more desired solution. At the beginning of the solution process, she utilized the possibility to adjust the r_i range to get familiar and work with the robustness measure. By considering robustness as an objective function, the DM learned how the robustness and base objective function values affect each other. Consequently, she utilized this knowledge to find a satisfactory balance between the robustness and the based objective function values of the final solution she accepted.

4.2 Procurement contract selection with pricing optimization for a process network

Next, we illustrate the application of our interactive approach by solving a problem in procurement contract selection with pricing optimization for a process network. We utilize the optimization model presented in Calfa and Grossmann (2015) as the foundation and augment it with three additional objectives. In the model, procurement contract selection and pricing analytics are combined for multi-period, multi-site tactical production planning. The manufacturer needs to make two key decisions: to select procurement contracts and to set selling prices for products. For the selection of procurement contracts, the manufacturer needs to decide whether to sign or not a particular contract with a supplier for purchasing a type of raw material. For setting the selling prices of final products, the manufacturer is assumed to use the price-response model (see e.g., Phillips 2005).

The problem was modeled with a single profit-focused objective in Calfa and Grossmann (2015). In this paper, we consider three additional objectives for environmental responsibilities and the maintenance of strategic competence of the manufacturer. The additional objectives include: minimizing the environmental impact scores of selected business partners (i.e., suppliers in this case), minimizing the pollution content emitted from the production process, and maximizing the demand in the market for the main products. Both the pollution content and environmental impact scores are for the consideration of environmental responsibility. Minimizing the pollution content emission is to improve the sustainability of the internal manufacturing process. Minimizing the environmental impact scores of suppliers aims at a responsible choice in business partners. Maximizing the demand in the market is to consolidate the strategic competence in the market.

The processing network considered has been presented in Section 5.1 of Calfa and Grossmann (2015). We consider a time horizon with a 3 months period and the manufacturer needed to decide whether or not to sign contracts with two suppliers with different bulk discount contracts and two suppliers with different discount contracts. Differing from Calfa and Grossmann (2015), where uncertainty due to future developments was incorporated as stochastic parameters, we consider the uncertainty in the production process. It results in perturbations of the amount of raw materials consumed, which affect the objectives of maximizing the profit and minimizing the pollution of the production process. The other two objectives do not involve uncertainty. All data used can be found in Calfa and Grossmann (2015) and in Appendix of this paper.

We solved this problem by applying our interactive approach with a real DM. For computing solutions, we used Gurobi® to solve the mixed integer quadratic problem after scalarization. We first calculated the ideal objective vector $z^* = (28.25, 0, 0, 87.88)^T$. The nadir objective vector was approximated as $z^{nad} = (0, 36, 12.31, 0)^T$. To get started, we set the ideal value of the robustness measure as 0 and the nadir value as 1.

We initialized the solution process by first introducing our robustness measure and visualization to the DM. The DM specified that the consumed raw materials in the product process can vary by 8% of their base values. Then we computed and presented an initial solution $z^0 = (28.25, 36.0, 6.22, 53.64, 0.08)^T$. The solution is visualized with our visualization method as illustrated in Fig. 10. In the figure, the objective function value of maximizing the profit under uncertainty can exceed its deterministic ideal value. The purpose of presenting the ideal and nadir values of the objective functions is to help the DM to get a general information on the ranges of the values of the base solution. Based on the initial solution, the DM wanted to decrease the environmental score of selected suppliers to 30 and increase the market demand of the main products to 75. At the same time, he also wanted to keep the pollution of the production process at its current value and allow the profit and the robustness of the solution to be impaired until 22 and 0.15 respectively.

Based on his preferences, a new solution was calculated and presented to him. As the interactive solution process is described in Sect. 4.1 in details, we here omit the detailed description to avoid repetitions. Instead, we summarize the

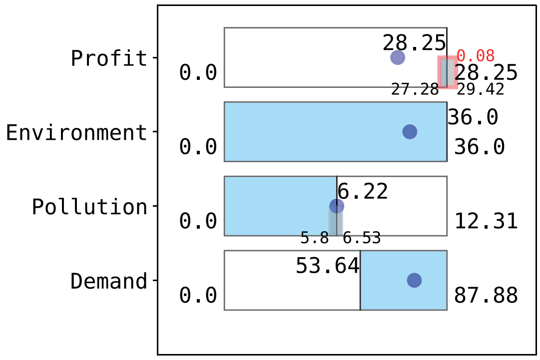


Fig. 10 Intial solution

preferences of the DM and the objective function values of solutions computed in each iteration in Table 2.

After four iterations, the final solution was satisfactory. During the solution process, the DM observed that the first objective (maximizing the profit) was more sensitive to the uncertainty in the production process than the third objective (minimizing the pollution content emitted). The DM understood that this is a property of the problem. This problem is a mixed-integer optimization problem and only the raw materials consumed which are continuous decision variables involved uncertainty. With the help of the suitable solver, our approach was able to handle the preferences of the DM and find solutions accordingly. In addition, our approach helped the DM to understand the consequences of the involved uncertainty and thus, supported him to consider the objective function values and the robustness of solutions simultaneously.

Table 2 Iterations of the interactive solution process

Iteration	Solution	Preferences
Initial	$z^0 = (28.25, 36.0, 6.22, 53.64, 0.08)$	$I^{\geq} = \{1, 5\}, \epsilon = \{22, 0.15\},$ $I^{=}\{3\}, I^{\leq} = \{2, 4\}, \hat{z} = \{30, 75\}$
1	$z^1 = (22.12, 2.0, 6.22, 53.91, 0.073)$	$I^{<} = \{3, 5\}, I^{\geq} = \{1\}, \epsilon_1 = 20$
2	$z^2 = (20.0, 5.0, 3.04, 28.29, 0.05)$	$I^{<} = \{4\}, I^{\geq} = \{1, 2, 3, 5\},$ $\epsilon = \{20, 30, 9.5, 0.15, \}$
3	$z^3 = (20.0, 13.33, 9.50, 74.48, 0.12)$	$I^{\geq} = \{1, 2, 4, 5\},$ $\epsilon = \{20, 30, 70, 0.15\}, I^{\leq} = \{3\}, \hat{z}_3 = 8$
4	$z^4 = (20.0, 2.0, 8.71, 70.0, 0.10)$	$I^{\geq} = \{1, 2, 4, 5\},$ $\epsilon = \{20, 30, 60, 0.15\}, I^{\leq} = \{3\}, \hat{z}_3 = 7.5$
5	$z^5 = (18.23, 2.0, 7.11, 60.0, 0.095)$	–

5 Conclusions

In this paper, we focused on supporting the DM to simultaneously consider the objective function values and robustness of solutions for multiobjective optimization problems with decision uncertainty. Based on the desired properties for a robustness measure to be used in an interactive approach, we introduced a new robustness measure that can deliver the meaning of robustness in an understandable way to the DM.

We proposed an interactive approach by utilizing elements of the synchronous NIMBUS method which is specifically suitable for solving problems with decision uncertainty. Because of the incorporation of robustness, we modified two components of the interactive NIMBUS solution process. We tailored a visualization method specifically for the new robustness measure and the associated robustness information by superimposing them on top of the bars representing the original objective functions. With this visualization, we can help the DM to consider the objective function values and the robustness of a solution at the same time. We also added a step to provide multiple alternatives for the DM to specify her/his preferences on the robustness of a more desired solution. Even though we built our approach based on NIMBUS, same idea and robustness measure can be applied to other classification based and reference point based methods. We demonstrated the advantages of the interactive approach by solving the river pollution problem and the problem in procurement contract selection with price optimization in a process network. Naturally, this approach can also be used to solve a wider range of problems.

Since we made the information on the r_i ranges of all objective functions available, we can further allow the DM to directly specify preferences on robustness for all or selected objectives in the future. As some of the objectives might be more important in considering robustness than others, we can incorporate the information about the importance also into our robustness measure. Also, as in Miettinen et al. (2014), the obtained solution can be further analyzed to quantify how much worse the solutions are compared to the Pareto optimal solutions of the original problem.

Acknowledgements We thank Dr. Dmitry Podkopaev for various discussions in constructing the multiobjective version of the procurement contract selection and pricing optimization problem.

Appendix: data for the example problem in Sect. 4.2

The model of the problem solved is based on Calfa and Grossmann (2015). Originally, it has only one objective to maximize profit. We augmented the model with three additional objectives: minimizing the environmental factors, minimizing the emission of pollutant, and maximizing the market demand of the main product. The objective to maximize profit (f_1 in this paper) and the estimation on the demand (f_4 in this paper) as well as the related data can be found in Calfa and Grossmann (2015). The objective for responsible selection of suppliers has been inspired by Yeh and Chuang (2011). Using the same notation as in Calfa and Grossmann (2015), we have

$$f_2(x) = \frac{1}{T} \sum_{t=1}^{T} \sum_{q=1}^{Q} \sum_{s=1}^{S} G^q_{s,t} y_{s,t}.$$

In the equation, $G^q_{s,t}$ represents the q-th environmental impact score of the supplier s in the planning period t and $y_{s,t}$ is the binary decision variable representing whether the supplier s is selected in the period t. The objective function is to take the average of the aggregation of all the environmental impact scores of all selected suppliers in each period.

For the consideration of pollutant emission, we consider the amount of sulphur dioxide emitted to the air based on the amount of sulphur content in the purchased raw materials. We have

$$f_3(x) = \frac{1}{T} \sum_{t=1}^{T} \sum_{j=1}^{P} \sum_{c=1}^{M} E_{j,t} W^c_{j,t}.$$

The emission factor of the process plant j in th time period t is represented by $E_{j,t}$. The notation $W^c_{j,t}$ is the amount of raw material c consumed in the plant j in the period t. The emission factor of the plants in different time periods can be different due to variation of the heating material used. In f_2 and f_3, $G^q_{s,t}$ and $E_{j,t}$ are parameters.

The environmental impact score $G^1_{s,t}$ depends on percentage the supplier s has been paying attention to the environmental protection policies in the time period t:

$$G^1_{s,t} = \begin{cases} 1 : & 100\% \\ 2 : & \text{more than } 50\% \\ 3 : & \text{less than } 50\% \\ 4 : & \text{none.} \end{cases}$$

The score $G^2_{s,t}$ depends on the percentage of sustainability of the product of the supplier s in the time period t:

$$G^2_{s,t} = \begin{cases} 1 : & 100\% \\ 2 : & \text{more than } 50\% \\ 3 : & \text{less than } 50\% \\ 4 : & \text{none.} \end{cases}$$

The score $G^3_{s,t}$ depends on the percentage of green customers' market share of the supplier s in the time period t:

$$G^3_{s,t} = \begin{cases} 1: & \text{above } 80\% \\ 2: & 60\% \text{ to } 80\% \\ 3: & 40\% \text{ to } 60\% \\ 4: & 20\% \text{ to } 40\% \\ 5: & \text{less than } 20\%. \end{cases}$$

The score $G^4_{s,t}$ depends on the percentage of recycling product design of the supplier s in the time period t:

$$G^4_{s,t} = \begin{cases} 1: & 100\% \\ 2: & \text{more than } 50\% \\ 3: & \text{less than } 50\% \\ 4: & \text{none.} \end{cases}$$

Table 3 Emission factors of processing plants

Plant	Time period		
	1	2	3
p_1	0.22	0.3	0.24
p_2	0.15	0.18	0.24
p_3	0.21	0.17	0.22

Table 4 Environmental impact scores of supplier 1 (discount contract)

Scores	Time period		
	1	2	3
G^1	3	2	1
G^2	3	2	1
G^3	2	1	2
G^4	1	4	2

Table 5 Environmental impact scores of supplier 2 (bulk discount contract)

Scores	Time period		
	1	2	3
G^1	2	3	2
G^2	3	2	1
G^3	3	1	1
G^4	1	1	2

Table 6 Environmental impact scores of supplier 3 (discount contract)

Scores	Time period		
	1	2	3
G^1	4	1	3
G^2	3	3	4
G^3	1	2	2
G^4	2	3	1

Table 7 Environmental impact scores of supplier 4 (bulk discount contract)

Scores	Time period		
	1	2	3
G^1	1	2	2
G^2	3	2	3
G^3	4	3	1
G^4	4	4	4

Table 8 Discount contract suppliers

Supplier	Price	Discount price	Threshold
S_1	3.15	2.47	20
S_3	3.15	2.58	20

Table 9 Bulk discount contract suppliers

Supplier	Price	Discount price	Threshold
S_2	3.06	2.38	40
S_4	2.95	2.55	40

We used data presented in Table 3 for the emission factors of the processing plants. The scores of the four candidate suppliers in our problem setting are given in Tables 4, 5, 6, and 7, and the settings of the contracts are given in Tables 8 and 9.

References

Asafuddoula M, Singh H, Ray T (2015) Six-sigma robust design optimization using a many-objective decomposition-based evolutionary algorithm. IEEE Trans Evol Comput 19(4):490–507

Azaron A, Brown K, Tarim S, Modarres M (2008) A multi-objective stochastic programming approach for supply chain design considering risk. Int J Prod Econ 116(1):129–138

Barrico C, Antunes C (2006) Robustness analysis in multi-objective optimization using a degree of robustness concept. Proc IEEE Congr Evol Comput CEC 2006:1887–1892

Bertsimas D, Sim M (2004) The price of robustness. Oper Res 52(1):35–53
Bokrantz R, Fredriksson A (2017) Necessary and sufficient conditions for Pareto efficiency in robust multiobjective optimization. Eur J Oper Res 262(2):682–692
Branke J, Deb K, Miettinen K, Słowiński R (eds) (2008) Multiobjective optimization: interactive and evolutionary approaches. Springer, Berlin
Calfa B, Grossmann I (2015) Optimal procurement contract selection with price optimization under uncertainty for process networks. Comput Chem Eng 82:330–343
Deb K, Gupta H (2006) Introducing robustness in multi-objective optimization. Evol Comput 14(4):463–494
Ehrgott M, Ide J, Schöbel A (2014) Minmax robustness for multi-objective optimization problems. Eur J Oper Res 239(1):17–31
Eichfelder G, Krüger C, Schöbel A (2017) Decision uncertainty in multiobjective optimization. J Glob Optim 69(2):485–510
Gaspar-Cunha A, Covas JA (2007) Robustness in multi-objective optimization using evolutionary algorithms. Comput Optim Appl 39(1):75–96
Gunawan S, Azarm S (2005) Multi-objective robust optimization using a sensitivity region concept. Struct Multidiscip Optim 29(1):50–60
Hassanzadeh E, Nemati H, Sun M (2013). Robust optimization for multiobjective programming problems with imprecise information. Proc Comput Sci 17:357–364. First International Conference on Information Technology and Quantitative Management
Ide J, Schöbel A (2016) Robustness for uncertain multi-objective optimization: a survey and analysis of different concepts. OR Spectr 38(1):235–271
Li M, Azarm S, Aute V (2005) A multi-objective genetic algorithm for robust design optimization. In : Proceedings of the 7th annual conference on genetic and evolutionary computation, GECCO 2005, pp 771–778
Liang Y, Cheng X, Li Z, Xiang J (2011) Robust multi-objective wing design optimization via CFD approximation model. Eng Appl Comput Fluid Mech 5(2):286–300
Miettinen K (1999) Nonlinear multiobjective optimization. Kluwer Academic Publishers, Boston
Miettinen K (2006) IND-NIMBUS for demanding interactive multiobjective optimization. In: Trzaskalik T (ed) Multiple criteria decision making '05. The Karol Adamiecki University of Economics in Katowice, Katowice, pp 137–150
Miettinen K, Mäkelä MM (2002) On scalarizing functions in multiobjective optimization. OR Spectr 24(2):193–213
Miettinen K, Mäkelä MM (2006) Synchronous approach in interactive multiobjective optimization. Eur J Oper Res 170(3):909–922
Miettinen K, Mustajoki J, Stewart TJ (2014) Interactive multiobjective optimization with NIMBUS for decision making under uncertainty. OR Spectrum 36(1):39–56
Narula S, Weistroffer H (1989) A flexible method for nonlinear multicriteria decision-making problems. IEEE Trans Syst Man Cybern 19(4):883–887
Ojalehto V, Miettinen K, Laukkanen T (2014) Implementation aspects of interactive multiobjective optimization for modeling environments: the case of GAMS-NIMBUS. Comput Optim Appl 58(3):757–779
Phillips R (2005) Pricing and revenue optimization. Stanford University Press, Palo Alto
Salimi A, Lowther DA (2016) On the role of robustness in multi-objective robust optimization: application to an IPM motor design problem. IEEE Trans Magn 52(3):1–4
Sawaragi Y, Nakayama H, Tanino T (1985) Theory of multiobjective optimization. Academic Press Inc., Cambridge
Steuer R (1986) Multiple criteria optimization: theory, computation and applications. Wiley, New York
Sun G, Li G, Zhou S, Li H, Hou S, Li Q (2010) Crashworthiness design of vehicle by using multiobjective robust optimization. Struct Multidiscip Optim 44(1):99–110
Talaei M, Moghaddam BF, Pishvaee MS, Bozorgi-Amiri A, Gholamnejad S (2016) A robust fuzzy optimization model for carbon-efficient closed-loop supply chain network design problem: a numerical illustration in electronics industry. J Clean Prod 113:662–673
Wiecek MM, Dranichak GM (2016) Robust multiobjective optimization for decision making under uncertainty and conflict. In: Gupta A, Capponi A, Smith JC, Greenberg HJ (eds) Optimization challenges in complex, networked and risky systems. Springer, New York, pp 84–114
Wiecek MM, Blouin VY, Fadel GM, Engau A, Hunt BJ, Singh V (2009) Multi-scenario multi-objective optimization with applications in engineering design. In: Barichard V, Ehrgott M, Gandibleux X,

T'Kindt V (eds) Multiobjective programming and goal programming: theoretical results and practical applications. Springer, Berlin, pp 283–298
Wierzbicki AP (1982) A mathematical basis for satisficing decision making. Math Model 3(5):391–405
Yeh W-C, Chuang M-C (2011) Using multi-objective genetic algorithm for partner selection in green supply chain problems. Expert Syst Appl 38(4):4244–4253

PII

INTERACTIVE MULTIOBJECTIVE ROBUST OPTIMIZATION WITH NIMBUS

by

Yue Zhou-Kangas, Kaisa Miettinen, and Karthik Sindhya 2018

In: M. Baum, G. Brenner, J. Grabowski, T. Hanschke, S. Hartmann, A.Schöbel (eds) Simulation Science. SimScience 2017. Communications in Computer and Information Science, vol 889, p60-76, Springer.

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

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

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:

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

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 $\max\limits_i[w_i(f_i(\boldsymbol{x'},\boldsymbol{\xi})-\bar{z}_i)]+\rho\sum\limits_{i=1}^{k}(f_i(\boldsymbol{x'},\boldsymbol{\xi})-\bar{z}_i)<\max\limits_i[w_i(f_i(\boldsymbol{x^*},\boldsymbol{\eta})-\bar{z}_i)]+\rho\sum\limits_{i=1}^{k}(f_i(\boldsymbol{x^*},\boldsymbol{\eta})-\bar{z}_i)$, where for all $\boldsymbol{\xi}\in\mathcal{U}$ there exists a $\eta\in\mathcal{U}$ which satisfy the inequality. Further, we know that $\max\limits_{\boldsymbol{\xi}\in\mathcal{U}}\max\limits_i[w_i(f_i(\boldsymbol{x'},\boldsymbol{\xi})-\bar{z}_i)]+\rho\sum\limits_{i=1}^{k}(f_i(\boldsymbol{x'},\boldsymbol{\xi})-\bar{z}_i)<$

$\max_{\boldsymbol{\eta}'\in\mathcal{U}} \max_i [w_i(f_i(\boldsymbol{x}^*,\boldsymbol{\eta}') - \bar{z}_i)] + \rho \sum_{i=1}^{k}(f_i(\boldsymbol{x}^*,\boldsymbol{\eta}') - \bar{z}_i)$. 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.

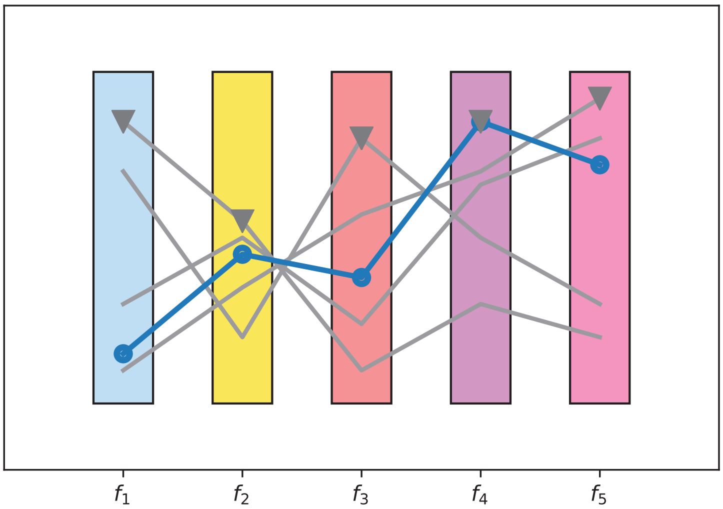


Fig. 1. Outcomes in the worst case

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}.

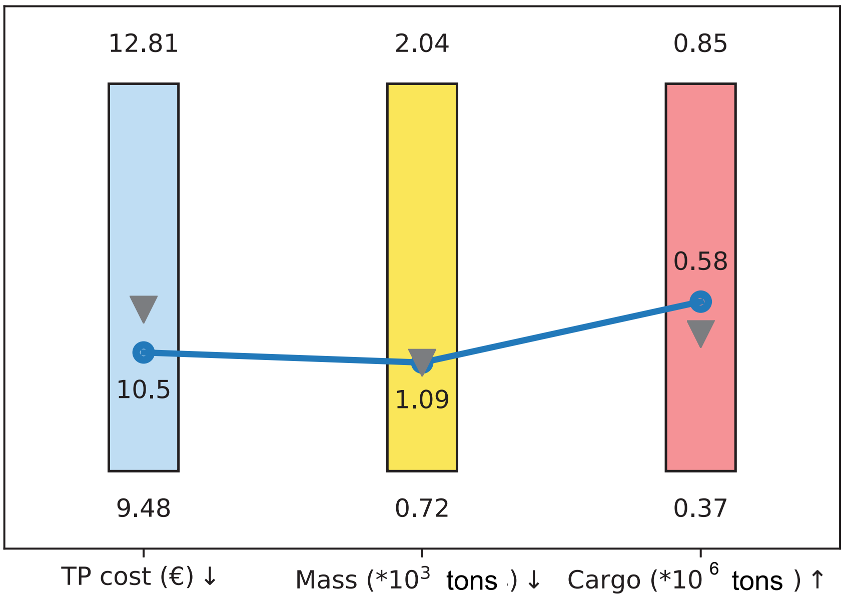


Fig. 2. Iteration 1 of ship design problem

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

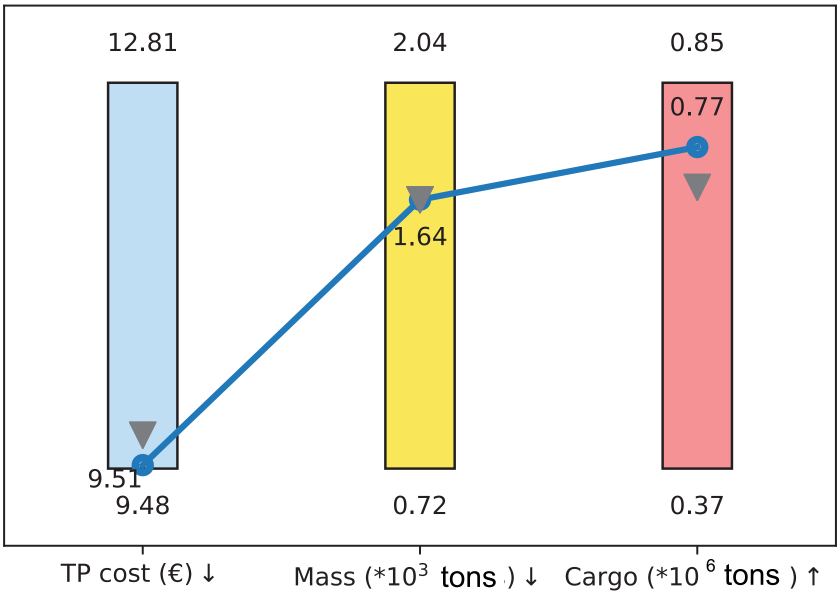


Fig. 3. Iteration 2 of ship design problem

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

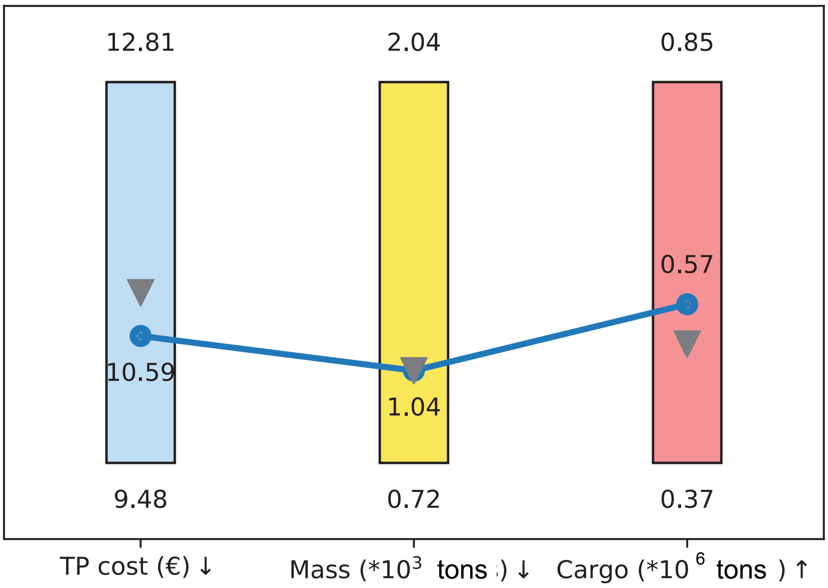


Fig. 4. Iteration 3 of ship design problem

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)

PIII

A SIMPLE INDICATOR BASED EVOLUTIONARY ALGORITHM FOR SET-BASED MINMAX ROBUSTNESS

by

Yue Zhou-Kangas and Kaisa Miettinen 2018

In A. Auger, C. M. Fonseca, N. Lourenco, P. Machado, L. Paquete, and D. Whitley (eds) Parallel Problem Solving from Nature - PPSN XV. PPSN 2018. Lecture Notes in Computer Science, vol 11101, p286-297 Springer.

A simple indicator based evolutionary algorithm for set-based minmax robustness

Yue Zhou-Kangas and Kaisa Miettinen

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

Abstract. For multiobjective optimization problems with uncertain parameters in the objective functions, different variants of minmax robustness concepts have been defined in the literature. The idea of minmax robustness is to optimize in the worst case such that the solutions have the best objective function values even when the worst case happens. However, the computation of the minmax robust Pareto optimal solutions remains challenging. This paper proposes a simple indicator based evolutionary algorithm for robustness (SIBEA-R) to address this challenge by computing a set of non-dominated set-based minmax robust solutions. In SIBEA-R, we consider the set of objective function values in the worst case of each solution. We propose a set-based non-dominated sorting to compare the objective function values using the definition of lower set less order for set-based dominance. We illustrate the usage of SIBEA-R with two example problems. In addition, utilization of the computed set of solutions with SIBEA-R for decision making is also demonstrated. The SIBEA-R method shows significant promise for finding non-dominated set-based minmax robust solutions.

Keywords: minmax robust Pareto optimal solutions, hypervolume, set-based dominance, SIBEA, uncertainty

1 Introduction and background

The need to simultaneously consider multiple objectives and the existence of uncertainty from various sources complicate real-world optimization problems. Uncertainty due to for example imprecise data or uncertain future developments usually reflects as parameters in the objective functions. Traditional multiobjective optimization methods concentrate on optimizing multiple objectives simultaneously and finding a set of Pareto optimal or non-dominated solutions for deterministic formulations of problems. Different approaches can be used to find this set, for example with scalarization techniques (see e.g., [21]) or with evolutionary multiobjective optimization methods (see e.g., [8]). However, the involved uncertainty can affect deterministic Pareto optimal or non-dominated solutions with undesired degradation in their objective function values. Thus, considering uncertainty in the optimization process is as important as optimizing multiple objectives simultaneously.

The goal of handling uncertainty and multiple objectives simultaneously is finding robust solutions that are sufficiently immune to the uncertainty and with trade-offs among the objectives. Different concepts of robustness and measures of robustness have been proposed in the literature. Typically, robustness measures are incorporated into evolutionary multiobjective optimization methods to quantify the effects of uncertainty on the objective function values (e.g., [4, 9, 12, 17]). Different robustness concepts alter the definition of dominance. Based on the concepts, uncertain multiobjective optimization problems can be transformed to deterministic ones (as summarized in [14] and [25]). In addition, different possible values of uncertain parameters can be considered simultaneously during the optimization process (as e.g., in [22] and [24]).

Among the robustness concepts, the most widely used ones belong to the family of minmax robustness (e.g., [5, 11, 16]). Due to different possible values of the uncertain parameters, a solution in the decision space can correspond to a set of outcomes (i.e., objective function values). We refer to a set of outcomes corresponding to a solution as the outcome set of the solution. Minmax robustness compares the worst outcomes in the outcome sets and finds the best possible ones. The worst outcomes are referred to as the worst case outcome set.

Set-based minmax robustness [11] finds the solutions with the best worst case outcome sets by utilizing set-based dominance [23]. For feasible solutions considered, we need to identify their worst case outcome sets by maximizing the multiple objectives simultaneously in their outcome sets and compare them with set-based dominance. This series of tasks makes the computation of set-based minmax robust solutions challenging. Methods from robust optimization and mathematical optimization can only address the challenge partially.

Some solution methods via scalarizing and reformulating the scalarized subproblems have been proposed e.g., in [5, 16]. However, typically the reformulations are based on some (strict) assumptions on the characteristics of the problem which cannot be always guaranteed in practical problems. If no assumptions on the characteristics can be made, using samples to replace the uncertainty set has been explored e.g., in [27]. The shortcoming is that the resulting solutions might not be or near to minmax robust. The needs of obtaining a more accurately approximated set of set-based minmax robust solutions have motivated us for further developments.

Different types of evolutionary multiobjective optimization methods have been able to approximate solutions for many challenging problems. For comparing worst case outcome sets, methods which combine non-dominated sorting and crowding distance are not suitable since defining the crowding distance between the worst case outcome sets is not possible. Decomposition based methods cannot be directly applied since we cannot directly associate worst case outcome sets to the weighting vectors. Set-based dominance has been utilized in the evolutionary multiobjective optimization community e.g., in [3, 30]. The population is treated a whole set and set-based dominance is used to improve the population. Very recently, using set-based dominance to solve problems involving uncertainty has also attracted interest. In [15], a genetic algorithm has been proposed for solving combinatorial bi-objective optimization problems with a set of discrete values of the uncertain parameters. In [13], an evolutionary algorithm has been proposed for solving problems with interval uncertainty (i.e., the uncertain parameters stem from some intervals) with reformulated objective functions. A specific definition of set-based dominance has been used to compare the worst case outcomes in [2]. These earlier research demonstrates potential to address the challenges.

In this paper, we propose utilizing an evolutionary multiobjective optimization approach SIBEA-R to tackle the challenge of approximating set-based minmax robust Pareto optimal solutions. We extend SIBEA [28] for this purpose. We incorporate the definition of set-based minmax robustness into the SIBEA method and develop a non-dominated sorting procedure based on the lower set less order. We also utilize the hypervolume of the worst case outcome sets in the environmental selection process.

The rest of the paper is organized as follows: Section 2 presents some concepts we use in this paper. Section 3 presents SIBEA-R followed by some numerical examples of how it can be used in Section 4. Finally, Section 5 concludes the paper and identifies some future research directions.

2 Preliminaries

In this paper, we consider multiobjective optimization problems with uncertainty reflected in the parameters of the objective functions in the following form:

$$\begin{pmatrix} \text{minimize} & \left(f(x,\xi) = f_1(x,\xi), \cdots, f_k(x,\xi)^T\right) \\ \text{subject to} & x \in \mathfrak{X} \end{pmatrix}_{\xi \in \mathcal{U}}, \tag{1}$$

where $x = (x_1, \cdots, x_n)^T$ is the decision vector from the feasible set $\mathfrak{X}$ in the decision space $\mathbb{R}^n$ whose components are called decision variables and ξ consists of the uncertain parameters which are assumed to stem from an uncertainty set $\mathcal{U}$. With ξ stemming from $\mathcal{U}$, a solution $x \in \mathfrak{X}$ is mapped in the objective space as a set-valued map [23] under the objective functions $f_1, \cdots, f_k$ to the objective space. We call this set-valued map the outcome set and denote by $f_{\mathcal{U}}(x) = \{f(x,\xi), \xi \in \mathcal{U}\}$. In the outcome set, a specific objective vector $f(x,\xi)$ is called an outcome.

The set-based minmax robust counterpart of (1) is presented in [11] as:

$$\underset{x \in \mathfrak{X}}{\text{minimize}} \quad \underset{\xi \in \mathcal{U}}{\text{maximize}} \; f(x,\xi) = (f_1(x,\xi), \cdots, f_k(x,\xi))^T. \tag{2}$$

We say that a solution $x^* \in \mathfrak{X}$ is set-based minmax robust Pareto optimal for problem (1), if there does not exist another solution $x \in \mathfrak{X}$ such that $f_{\mathcal{U}}(x) \subseteq f_{\mathcal{U}}(x^*) - \mathbb{R}^k_{\geq}$, where $\mathbb{R}^k_{\geq} = \{x \in \mathbb{R}^k : x_i \geq 0, i = 1, \cdots, k\}$ [11]. This definition is based on the concept of lower set less order: let A and B be arbitrary closed sets, then $A \preceq^l B$ implies $A \subseteq B - \mathbb{R}^k_{\geq}$. Thus, when we compare two sets of vectors, we say $A \preceq^l B$ if for all $a \in A$ there exists $b \in B$ such that $a_i \leq b_i, i = 1, \cdots, k$.

Figure 1 illustrates an example of set-based minmax robustness with two objective functions to be minimized. In the example, we have a feasible set $\mathfrak{X} = \{x^1, x^2, x^3, x^4\}$ and an arbitrary uncertainty set $\mathcal{U}$. We plot the outcome set of the three solutions in the figure $f_{\mathcal{U}}(x^1)$ (bold solid curve), $f_{\mathcal{U}}(x^2)$ (bold dotted line), $f_{\mathcal{U}}(x^3)$ (bold dashed line), and $f_{\mathcal{U}}(x^4)$ (bold dash-dotted line). The gray thin lines help us to identify the borders of the outcome sets. Solution x^1 is a set-based minmax robust Pareto optimal solution, since $f_{\mathcal{U}}(x^1) - \mathbb{R}^2_{\geq}$ does not contain $f_{\mathcal{U}}(x^2)$ nor $f_{\mathcal{U}}(x^3)$. Similarly, we can see that x^2 and x^3 are also set-based minmax robust Pareto optimal solutions. However, x^4 is not set-based minmax robust Pareto optimal since $f_{\mathcal{U}}(x^4) - \mathbb{R}^2_{\geq}$ contains $f_{\mathcal{U}}(x^1)$ and $f_{\mathcal{U}}(x^3)$. The formulation (2) minimizes the worst case outcomes. As mentioned before,

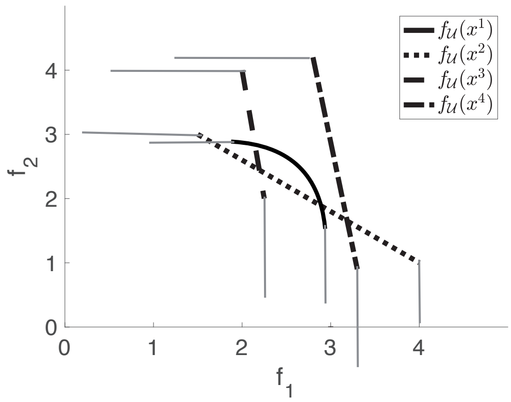


Fig. 1: Example of set-based minmax robustness

we need to first find the worst case outcomes and compare them as a whole. So, finding set-based minmax robust Pareto optimal solutions requires us to address these two challenges in a systematical way. Finding the worst case outcome set of a fixed solution $x \in \mathfrak{X}$ requires solving a multiobjective optimization problem with the objective functions to be maximized as follows:

$$\underset{\xi \in \mathcal{U}}{\text{maximize}} \quad (f_1(x, \xi), \cdots, f_k(x, \xi))^T. \tag{3}$$

3 The SIBEA-R method

In this section, we introduce SIBEA-R for approximating set-based minmax robust Pareto optimal solutions. We first introduce the steps of SIBEA-R. Then, we discuss details of the steps with a concentration on the further developments on SIBEA for set-based minmax robustness.

The SIBEA-R method takes the population size (NP) and the number of generations (NG) as the input and produces a set of non-dominated set-based minmax robust solutions A as the output. The basic steps are as follows:

Step 1. (Initialization) Generate an initial set of decision vectors P of size NP and find their worst case outcome sets by solving (3). Set the generation counter $m = 1$.
Step 2. (Mating) Create an offspring population Q using crossover and mutation operators and find their worst case outcome sets. Set $P = P \cup Q$.
Step 3. (Environmental selection) Rank the population P using lower set less order and sort the individuals into different fronts $F^i, i = 1, 2, \cdots$. and do the following:

- Set a new population $P^1 = \emptyset$. Set $i = 1$ and $P^1 = P^1 \cup F^1$. As long as $|P^1| < NP$, set $i = i + 1$, $P^1 = P^1 \cup F^i$. The notation $|P^1|$ represents the cardinality of P^1.
- if $|P^1| = NP$, set $P = P^1$ and go to Step 4. Otherwise, do the following until $|P^1| = NP$: identify the solutions with the worst rank $P' \subset P^1$.
- For each solution $x \in P'$, determine the loss of the value of the hypervolume indicator $d(x)$ if it is removed from the set P'. Remove the solution with the smaller loss from P', i.e., set $P' = P' \setminus \{x\}$

Step 4. (Termination) If $m > NG$, set $A = P^1$ and stop. Otherwise, set $m = m + 1$ and go to Step 2.

In Steps 1 and 2, we consider the worst case outcome sets of the individuals and their offspring. We have mentioned earlier that for a fixed solution, finding its worst case outcomes is a multiobjective optimization problem with objectives to be maximized in the uncertainty set. We can solve the maximization problem with an evolutionary multiobjective optimization method to approximate a set of outcomes in the worst case. However, doing so requires a lot of computation resources. Thus, we should find a representative set of solutions of the maximization problem and use it to save the computation resource.

We propose to systematically solve a small number of scalarized subproblems to obtain the representative worst case outcome sets. For example, we can utilize the approach used in [6] to generate a set of evenly distributed points on a unit hyperplane in the objective space. Then, we use them as the reference points to optimize a series of the achievement scalarizing functions (see e.g., [26]). In what follows we denote the number of worst case outcomes in the representative worst case outcome set by W and the values of the uncertain parameters which the objective functions reach their worst case values by $\xi^w, w = 1, \cdots, W$. The number of function evaluations depends on the solver used to solve the scalarized subproblems. In case of discrete scenarios in the uncertainty set, the number of function evaluations is $k \times NP \times NG\times$ number of scenarios.

After we have found the representative worse case outcome sets of the individuals, we need to rank them and sort them into different fronts. We call this step set-based non-dominated sorting, where we define the dominance between two representative worst case outcome sets with lower set less order. The sorting procedure is inspired by that presented in [10]. The steps of the set-based non-dominated sorting are as follows:

Step 1. For each solution $p \in P$, set the domination count $n_p = 0$ and the set of solutions dominated by p as an empty set $S_p = \emptyset$. Set $P = P \setminus \{p\}$ and carry out the following steps:

(a) For each $q \in P$, do the following:
If for all $f(q, \xi^w), w = 1, \cdots, W$, there exists $f(p, \xi^w)$ such that $f(q, \xi^w) \leq f(p, \xi^w)$, set $n_p = n_p + 1$.
Otherwise if for all $f(p, \xi^w), w = 1, \cdots, W$, there exists $f(q, \xi^w)$ such that $f(p, \xi^w) \leq f(q, \xi^w)$, set $S_p = S_p \cup \{q\}$

(b) If $n_q = 0$, then $p^{rank} = 1$ and $F^1 = F^1 \cup \{p\}$.

Step 2. Set front counter i = 1

Step 3. Do the following steps until $F^i = \emptyset$

For each $p \in F^i$
for each $q \in S_p$
set $n_q = n_q - 1$
if $n_q = 0$, then $q^{rank} = i+1$, and $F^{i+1} = F^{i+1} \cup \{q\}$, set $i = i+1$ and continue with Step 3 to the next front.

In the set-based non-dominated sorting, Step 2(a) is for checking if $f_{\mathcal{U}}(p) \preceq^l f_{\mathcal{U}}(q)$ or $f_{\mathcal{U}}(q) \preceq^l f_{\mathcal{U}}(p)$. We pair-wise compare the solutions and go through the outcomes in the representative worst case outcome sets.

After we have sorted the solutions into different fronts, we start the environmental selection in Step 3. We fill the next generation population incrementally starting from solutions that are in F^1 until the number of solutions exceeds the population size NP. Then we delete the solutions from the last front based on the loss of the value of the hypervolume indicator (see e.g., [1] and [28]). We calculate the loss of the hypervolume when deleting a solution x' as $d(x') = H(S) - H(S')$, where $S = \{\tilde{f_{\mathcal{U}}}(x) : x \in P'\}$ and $S' = S \setminus \{\tilde{f_{\mathcal{U}}}(x')\}$. Here, we use $\tilde{f_{\mathcal{U}}}$ instead of $f_{\mathcal{U}}$ because we consider the representative worst case outcome sets.

After step 3, we have a new population. If the number of generations has been exceeded, we terminate the solution process and take the set-based non-dominated solutions of the last generation as the output set A. If the number of generations has not been exceeded, we continue by going to Step 2.

After obtaining the set A, a decision maker should choose a final solution. For example, [27] uses an interactive post-processing procedure to find the final solution based on preference information. In the interactive process, we present the outcome of a solution in the nominal case which is the undisturbed or usual case. Then, the decision maker can specify her or his preferences for a more desired solution until (s)he finds a satisfactory solution. The purpose is to help the decision maker to find the final solution based on the nominal value and at the same time the solution is the best possible when the worst case happens.

4 Numerical results

In this section, we demonstrate the usage of the SIBEA-R method with two example problems. The examples help us to test our proposal of using set-based non-dominated sorting in an evolutionary algorithm. The first example problem is a simple linear problem based on one of the examples presented in [25]:

$$\left(\begin{array}{ll} \text{minimize} & \begin{pmatrix} 2\xi_1 x_1 - 3\xi_2 x_2 \\ 5\xi_1 x_1 + \xi_2 x_2 \end{pmatrix} \\ \text{subject to} & 0 \le x_1 \le 1.5 \\ & 0 \le x_2 \le 3 \end{array}\right)_{\xi \in \mathcal{U}}, \tag{4}$$

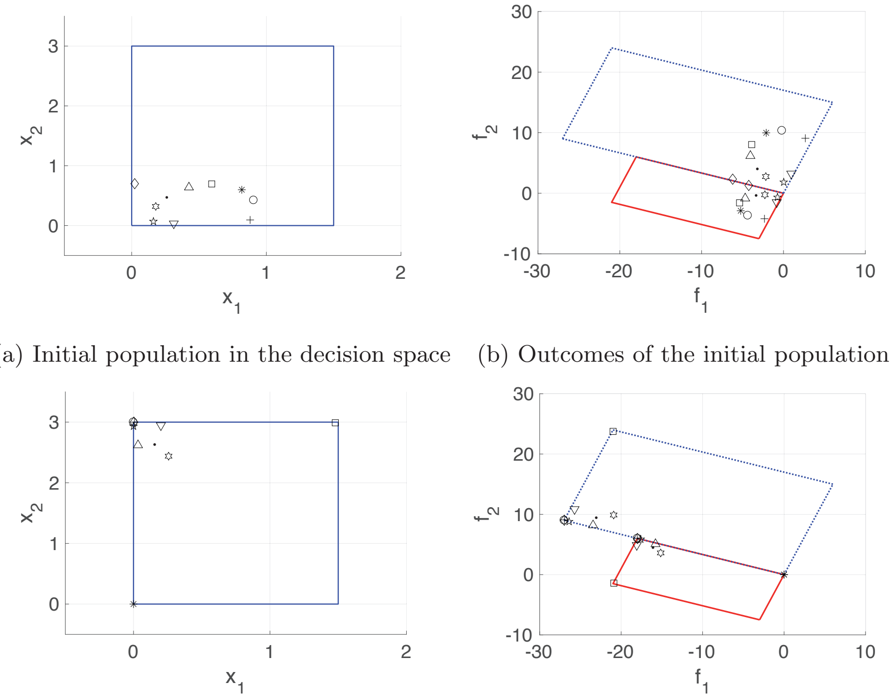
where $\mathcal{U} = \left\{ \left(\begin{smallmatrix} -1 \\ 2 \end{smallmatrix} \right), \left(\begin{smallmatrix} 2 \\ 3 \end{smallmatrix} \right) \right\}$.

In the experiments, we used the default setting of parameters as in the implementation of SIBEA in [7]. For (4), we can compute the outcomes in both possible sets of values for the uncertain parameters. We first illustrate the evolvement of the population, we visualize the initial generation in the decision space in Figure 2a and in the objective space in Figure 2b. In the figures, the solid lines are the borders of the feasible set and we visualize 10 individuals because of limited varieties of markers. In Figure 2b, the same marker appears twice because of the two possible cases in $\mathcal{U}$. We use SIBEA-R to evolve the population by considering their outcome sets (each set consists of two outcomes with the same marker in the figure). After 100 generations, the last generation is shown in Figure 2c in the decision space and in Figure 2d in the objective space.

We then studied the final populations of 20 independent runs with $NP = 30$. It is not even possible to compute a complete set of set-based robust Pareto optimal solutions for linear problems like (4). To the best of our knowledge, methods with similar ideas in the literature (e.g., [2]) had a different definition of robust Pareto optimality. We cannot easily benchmark the example problems. Thus, we first visually compare the solutions computed by SIBEA-R with 30 solutions computed by the weighted-sum approach proposed in [11]. The purpose is to use the solutions computed by the weighted-sum approach as references.

Figures 3a and 3b illustrate the solutions computed by the weighted-sum approach and SIBEA-R. The solutions computed by the weighted-sum approach are marked as solid red circles in the figures and the solutions computed by SIBEA-R are marked by the gray plus sings. In the figures, the gray cloud consists of the solutions computed with 20 runs of the SIBEA-R method. We can see that SIBEA-R was able to find the solutions found by the weighted-sum approach. In addition, SIBEA-R also found other solutions in the interior of the feasible space. The existence of set-based minmax Pareto optimal solutions in the interior of the feasible space is proven in [20]. For example, the point $(0.5, 2.4)$ is set-based minmax robust Pareto optimal which can be checked by the definition. Based on the visualizations, we can observe that SIBEA-R has considered the outcomes concerning both sets of possible values of the uncertain parameters and found a set of non-dominated set-based minmax robust solutions.

The second example problem is based on a standard benchmark problem, ZDT2 (see, e.g., [8]). In this problem, we introduced two uncertain parameters which stem from a polyhedral uncertainty set. A polyhedral uncertainty set is given as the convex hull of a finite set of points. Even though modifying the problem can cause the loss of the characteristics of the carefully designed test



(a) Initial population in the decision space (b) Outcomes of the initial population

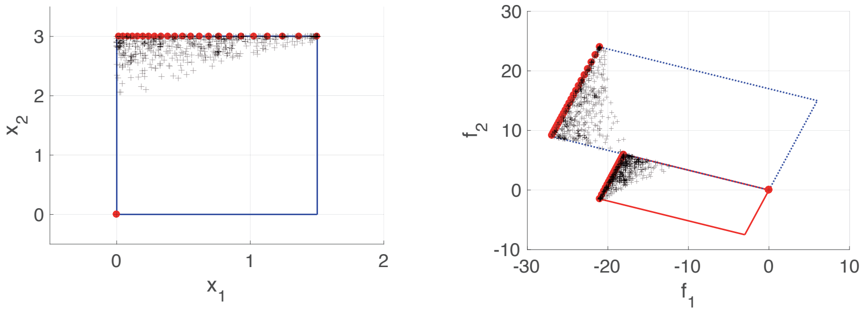
(c) Final population in the decision space (d) Outcomes of the final population

Fig. 2: The evolvment of the population by SIBEA-R

problems, our purpose is to illustrate the solutions founds by SIBEA-R and the usage of them for decision making. For the ZDT2-based problem, we set $NG = 100, NP = 30$ and found six worst case outcomes to represent the worst case outcome set. We run SIBEA-R 20 times to solve the problem.

We analyzed the results with the so-called average non-dominated objective space (i.e., the percentage of the volume of objective space between the ideal point and a reference vector which are not covered by the solutions) in each generation in all the runs to observe the convergence (see details in [29]). We also analyzed the attainment surface of the worst case outcome sets from multiple runs with the empirical attainment function graphical tools [18, 19]. We visualized the $25\%, 50\%, 75\%$ attainment surfaces.

The average non-dominated objective space in each generation for the 20 runs of the ZDT2-based problem is illustrated in Figure 4. The figure shows that the non-dominated objective space gradually reduced with generations and at the final generations, the average non-dominated space stayed stable. This means that the objective function values of solutions reduced along the generations. The attainment surfaces of the results from the 20 runs are shown in Figure 5. The figure illustrates that the solutions tend to converge to the area bounded



(a) Solutions in the decision space (b) Solutions in the objective space

Fig. 3: Solutions computed by the weighted-sum approach and SIBEA-R

by the intervals $f_1 = [0.5, 0.8], f_2 = [0.2, 0.7]$. Based on the experiment results, we can observe that SIBEA-R was able to improve the populations with the generations and the final populations of different runs were similar.

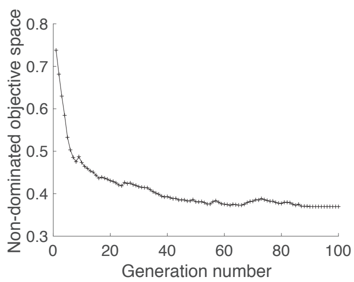


Fig. 4: Average non-dominated objective space, ZDT2-based problem

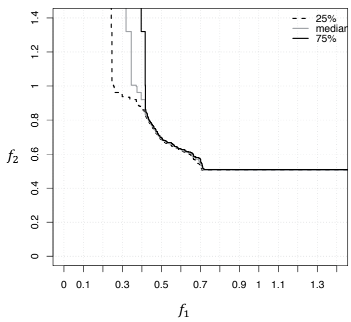


Fig. 5: Attainment surface, ZDT2-based problem

After SIBEA-R has found a set of non-dominated set-based minmax robust solutions, the set can be used for decision making. We illustrate the usage with a reference point-based interactive approach (see e.g., [21] for a detailed description). In a reference point-based approach, the decision maker specifies the desired objective function values as a reference point. We find a solution which satisfies the reference point as well as possible and present the solution to the decision maker. This kind of interactive process continues until the decision maker finds a most satisfactory solution. We used the final population of a run of the ZDT2-based problem and helped a decision maker to choose a final solution based on their outcomes in the nominal case. In the nominal case, the uncertain parameters behave normally without disturbance. So, we used the original ZDT2 problem as the nominal case. We carried out four iterations. The reference points and the solutions found are illustrated in Table 1. The solutions are also presented in Figure 6 with different markers. The decision maker took the third solution as the final solution since it is the nearest to her desired values.

ref.	solution	Marker
$(0.3, 0.7)^T$	$(0.43, 0.81)^T$	square
$(0.3, 0.95)^T$	$(0.3, 0.91)^T$	up triangle
$(0.5, 0.6)^T$	$(0.57, 0.67)^T$	diamond
$(0.8, 0.6)^T$	$(0.61, 0.61)^T$	down triable

Table 1: Interactive post-processing

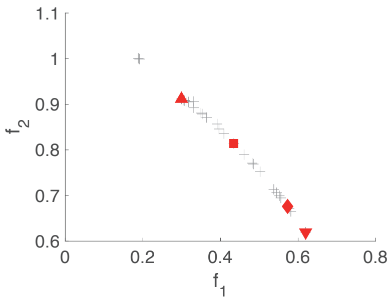


Fig. 6: Solutions found based on reference points

In the examples, we observed that SIBEA-R was able to find set-based minmax robust Pareto optimal solutions found by the weighted-sum approach. It was also able to find some solutions that the weighted-sum approach was not able to find. In the ZDT2-based problem, SIBEA-R was stable regarding finding similar final populations in different runs. These observations suggested that SIBEA-R has an appealing potential for approximating set-based minmax robust Pareto optimal solutions, which can be then used for decision making.

5 Conclusions

In this paper, we proposed SIBEA-R to compute an approximated set of set-based minmax robust Pareto optimal solutions. This is an initial study to explore opportunities evolutionary multiobjective optimization methods can provide in tackling challenges with robustness which are otherwise difficult. In SIBEA-R, instead of considering single outcomes, we considered the worst case outcome sets of solutions. We proposed a set-based non-dominated sorting procedure based on the lower set less order to rank the solutions for environmental selection. We illustrated the utilization of SIBEA-R with two example problems. The experiments on the example problems suggest that SIBEA-R can approximate set-based minmax robust Pareto optimal solutions. We also illustrated how the solutions found by SIBEA-R can be used in decision making.

Due to the set-based non-dominated sorting and the calculation of the hypervolume of outcome sets, SIBEA-R is computationally expensive and it tends to work with small population sizes. Thus, an immediate future research direction is to improve the computational efficiency and enable the calculation of a larger number of non-dominated set-based minmax robust solutions. In this paper, we only presented a limited amount of numerical experiments. It is necessary to extend the numerical experiments to a wider range of problems to further identify the strengths and limitations of SIBEA-R.

Acknowledgments

We thank Dr. Tinkle Chugh for useful discussions and providing an implementation of SIBEA. This research is related to Decision Analytics (DEMO).

References

1. A. Auger, J. Bader, D. Brockhoff, and E. Zitzler. Theory of the hypervolume indicator: Optimal μ-distributions and the choice of the reference point. In *Proceedings of the Tenth ACM SIGEVO Workshop on Foundations of Genetic Algorithms*, pages 87–102, New York, NY, USA, 2009. ACM.
2. G. Avigad and J. Branke. Embedded evolutionary multi-objective optimization for worst case robustness. In *Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation*, GECCO '08, pages 617–624. ACM, 2008.
3. J. Bader, D. Brockhoff, S. Welten, and E. Zitzler. On using populations of sets in multiobjective optimization. In M. Ehrgott, C. M. Fonseca, X. Gandibleux, J.-K. Hao, and M. Sevaux, editors, *Evolutionary Multi-Criterion Optimization, Proceedings*, pages 140–154. Springer, 2009.
4. J. Bader and E. Zitzler. Robustness in hypervolume-based multiobjective search. Technical report, TIK Report 317, 2010.
5. R. Bokrantz and A. Fredriksson. Necessary and sufficient conditions for Pareto efficiency in robust multiobjective optimization. *European Journal of Operational Research*, 262(2):682–692, 2017.
6. R. Cheng, Y. Jin, M. Olhofer, and B. Sendhoff. A reference vector guided evolutionary algorithm for many-objective optimization. *IEEE Transactions on Evolutionary Computation*, 20(5):773–791, 2016.
7. T. Chugh, K. Sindhya, J. Hakanen, and K. Miettinen. An interactive simple indicator-based evolutionary algorithm (i-sibea) for multiobjective optimization problems. In A. Gaspar-Cunha, C. Henggeler Antunes, and C. C. Coello, editors, *Evolutionary Multi-Criterion Optimization*, pages 277–291. Springer, 2015.
8. C. A. C. Coello, G. B. Lamont, and D. A. V. Veldhuizen. *Evolutionary Algorithms for Solving Multi-Objective Problems, 2nd edition*. Springer, 2006.
9. K. Deb and H. Gupta. Introducing robustness in multi-objective optimization. *Evolutionary Computation*, 14(4):463–494, 2006.
10. K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.
11. M. Ehrgott, J. Ide, and A. Schöbel. Minmax robustness for multi-objective optimization problems. *European Journal of Operational Research*, 239(1):17–31, 2014.
12. A. Gaspar-Cunha and J. A. Covas. Robustness in multi-objective optimization using evolutionary algorithms. *Computational Optimization and Applications*, 39(1):75–96, 2007.

13. D. Gong, J. Sun, and Z. Miao. A set-based genetic algorithm for interval many-objective optimization problems. *IEEE Transactions on Evolutionary Computation*, 22(1):47–60, 2018.
14. J. Ide and A. Schöbel. Robustness for uncertain multi-objective optimization: a survey and analysis of different concepts. *OR Spectrum*, 38(1):235–271, 2016.
15. D. Konur and H. Farhangi. Set-based min-max and min-min robustness for multiobjective robust optimization. In K. Coperich, E. Cudney, and H. Hembhard, editors, *Proceedings of the 2017 Industrial and Systmes Engineering Research Conference*, pages 1–6. Institute of Industrial and Systems Engineers, 2017.
16. D. Kuroiwa and G. M. Lee. On robust multiobjective optimization. *Vietnam Journal of Mathematics*, 40(2,3):305–317, 2012.
17. M. Li, S. Azarm, and V. Aute. A multi-objective genetic algorithm for robust design optimization. In *Proceedings of the 7th Annual Conference on Genetic and Evolutionary Computation, GECCO 2005*, pages 771–778, 2005.
18. M. López-Ibáñez. EAF graphical tools. `http://lopez-ibanez.eu/eaftools`.
19. M. López-Ibáñez, L. Paquete, and T. Stützle. Exploratory analysis of stochastic local search algorithms in biobjective optimization. In T. Bartz-Beielstein, M. Chiarandini, L. Paquete, and M. Preuss, editors, *Experimental Methods for the Analysis of Optimization Algorithms*, pages 209–222. Springer, 2010.
20. D. Majewski. Robust bi-objective linear optimization. Master's thesis, University of Göttingen, 2014.
21. K. Miettinen. *Nonlinear Multiobjective Optimization*. Kluwer Academic Publishers, Boston, 1999.
22. K. Miettinen, J. Mustajoki, and T. J. Stewart. Interactive multiobjective optimization with NIMBUS for decision making under uncertainty. *OR Spectrum*, 36(1):39–56, 2014.
23. L. Rodrguez-Marn and M. Sama. (λ,c)-contingent derivatives of set-valued maps. *Journal of Mathematical Analysis and Applications*, 335(2):974–989, 2007.
24. M. M. Wiecek, V. Y. Blouin, G. M. Fadel, A. Engau, B. J. Hunt, and V. Singh. Multi-scenario multi-objective optimization with applications in engineering design. In V. Barichard, M. Ehrgott, X. Gandibleux, and V. T'Kindt, editors, *Multiobjective Programming and Goal Programming: Theoretical Results and Practical Applications*, pages 283–298. Springer, 2009.
25. M. M. Wiecek and G. M. Dranichak. Robust multiobjective optimization for decision making under uncertainty and conflict. In A. Gupta, A. Capponi, J. C. Smith, and H. J. Greenberg, editors, *Optimization Challenges in Complex, Networked and Risky Systems*, pages 84–114, 2016.
26. A. P. Wierzbicki. A mathematical basis for satisficing decision making. *Mathematical Modelling*, 3(5):391–405, 1982.
27. Y. Zhou-Kangas, K. Miettinen, and K. Sindhya. Interactive multiobjective robust optimization with NIMBUS. In *Proceedings of Clausthal-Goettingen International Workshop on Simulation Science 2017*. Springer. To appear, 2018.
28. E. Zitzler, D. Brockhoff, and L. Thiele. The hypervolume indicator revisited: On the design of Pareto-compliant indicators via weighted integration. In S. Obayashi, K. Deb, C. Poloni, T. Hiroyasu, and T. Murata, editors, *Evolutionary Multi-Criterion Optimization*, pages 862–876. Springer, 2007.
29. E. Zitzler, M. Laumanns, and L. Thiele. SPEA2: Improving the strength Pareto evolutionary algorithm for multiobjective optimization. Technical report, TIK Report 103, 2002.
30. E. Zitzler, L. Thiele, and J. Bader. On set-based multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 14(1):58–79, 2010.

PIV

THE PRICE OF MULTIOBJECTIVE ROBUSTNESS: ANALYZING SOLUTION SETS TO UNCERTAIN MULTIOBJECTIVE OPTIMIZATION PROBLEMS

by

Yue Zhou-Kangas and Anita Schöbel

Submitted to a Journal

The price of multiobjective robustness: Analyzing solution sets to uncertain multiobjective problems

Yue Zhou-Kangas[1] and Anita Schöbel[2]

[1]University of Jyvaskyla, Faculty of Information Technology P.O.Box 35 (Agora), FI-40014, University of Jyvaskyla, Finland, yue.y.zhou-kangas@jyu.fi
[2]Institut für Numerische und Angewandte Mathematik, Georg-August Universität Göttingen, Germany, schoebel@math.uni-goettingen.de

Abstract

Defining and finding robust efficient solutions to uncertain multiobjective optimization problems has been an issue of growing interest recently. Different concepts have been published defining what a "robust efficient" solution is. Each of these concepts leads to a different set of solutions, but it is difficult to visualize and understand the differences between these sets. In this paper we develop an approach for comparing such sets of robust efficient solutions, namely we analyze their outcomes under the nominal scenario and in the worst case using the upper set-less order from set-valued optimization. Analyzing the set of nominal efficient solutions, the set of minmax robust efficient solutions and different sets of lightly robust efficient solutions gives insight into robustness and nominal quality of these sets of solutions. Among others we can formally prove that lightly robust efficient solutions are good compromises between nominal efficient solutions and minmax robust efficient solutions. In addition, we also propose a measure to quantify the price of robustness of a single solution. Based on the measure, we propose two strategies which can be used to support a decision maker to find solutions to a multiobjective optimization problem under uncertainty. All our results are illustrated by examples.
Keywords: multiobjective robust optimization, decision making, uncertainty, price of robustness

1 Introduction

More and more complex optimization problems are being solved in the modern society. These problems are characterized by multiple conflicting objectives and they almost inevitably involve uncertainty due to imprecise data, uncertain future developments, implementation uncertainty and so on. Multiobjective robust optimization is an evolving field specifically aiming at finding robust solutions that are sufficiently immune to uncertainty.

While robust optimization for single-objective optimization problems is well researched, the topic of robust multiobjective optimization is relatively young. For example, for single-objective optimization problems, [1] presents many results on the classical concept of strict robustness where the solutions are optimized in the worst case. Due to the conservativeness of strictly robust solutions, i.e., their objective function values can be very bad in other cases, less conservative robustness concepts than strict robustness have also been developed, see e.g., [15] for an overview. In recent years, different robustness concepts have been developed to take uncertainty into account also for multiobjective optimization problems, see [18, 27] for surveys of the many evolving robustness concepts.

The easiest way to handle uncertainty in the input parameters is to identify the so-called *nominal scenario*, which is the most typical, the undisturbed, or the expected scenario, and to solve the problem only for this case. This means that the uncertainty is ignored and one receives a standard multiobjective optimization problem. The resulting (Pareto-) efficient solutions are called *nominal efficient*. However, nominal efficient solutions may be very bad in terms of their objective function values when uncertain data does not behave as expected.

In order to take data uncertainty into account, different varieties of minmax robust efficiency, see e.g., [7, 10, 12, 21] have been proposed for multiobjective optimization problems under uncertainty. The idea of minmax robust efficiency is to optimize the objective function in the worst case over all scenarios. The resulting solutions are called *minmax robust efficient*. Unfortunately, it is often not clear how they can be computed, and they are rather conservative.

Conservatism of minmax robust solutions has been recognized for single-objective optimization problems for several years, which has led to different concepts to control the loss of quality (in the nominal case) in single-objective robust optimization, see e.g., [2, 5, 9, 13, 14, 22, 25], or [15] for a survey. For the same reason, also for multiobjective optimization problems, concepts have been refined in [16, 18, 20]. Here, we focus on the concept of lightly robust efficiency in which one looks for a robust solution still satisfying a given tolerable nominal quality.

While the relationships between different types of (robust) efficient solutions have been analyzed in [18], a comparison between their nominal quality, i.e., the objective function values in the nominal case, and robustness has not been investigated so far. In this paper, we propose to analyze the objective function values of a (robust efficient) solution under the nominal scenario and in the worst case (i.e., taking the most pessimistic view). This approach helps to understand and explain the meaning of robustness and hence may assist the decision maker to find a good balance between the nominal quality and the robustness. In our main result we show that lightly robust efficient solutions behave as intended: they are good compromises between minmax robust efficient and nominal efficient solutions, both with respect to the nominal scenario and with respect to the worst case.

For single-objective optimization problems, the trade-off between the nominal quality and the robustness of a solution has been analyzed in [9, 25]. In this paper we extend such an analysis to multiobjective optimization problems, i.e., we discuss how the trade-off between nominal quality and robustness can be quantified by providing a measure for the price of robustness. Note that some literature considers multiple scenarios of a single-objective optimization problem simultaneously, e.g., in [17], which results in a multiobjective optimization problem. In this paper, we discuss the loss of nominal quality and gain in robustness when considering the possible values of the uncertain parameters in the objective functions of multiobjective optimization problems simultaneously. We also provide two approaches utilizing the measure 'price of robustness' to find different types of solutions depending on the preferences of a decision maker. Analogously, we also quantify the gain in robustness. By combining the values of the two measures, the decision maker can decide if it is worth to increase the robustness of the solution. If not, the decision maker can decide to take the nominal efficient solution or find a lightly robust efficient solution. In this way, the decision maker can have helpful information to find a balance between nominal quality and robustness.

The remainder of the paper is organized as follows: Section 2 formally introduces nominal efficiency, minmax robust efficiency, and lightly robust efficiency. Section 3 analyzes the relationships among the three different kinds of solutions in the nominal scenario and in the worst case. Section 4 illustrates the results on some numerical examples, followed by Section 5 which introduces the price of robustness as a new measure and develops two strategies to assist decision making. Finally, Section 6 concludes the paper.

2 Nominal efficiency, minmax robust efficiency and lightly robust efficiency

Let a feasible set $\mathfrak{X}$, an uncertainty set $\mathcal{U}$ and a function $f : \mathfrak{X} \times \mathcal{U} \to \mathbb{R}^k$ be given. We deal with the following *uncertain problem*

$$(P_{\mathcal{U}}) \qquad \left(\begin{array}{ll} \min & f(x,\xi) := \begin{pmatrix} f_1(x,\xi) \\ f_2(x,\xi) \\ \vdots \\ f_k(x,\xi) \end{pmatrix} \\ \text{s.t.} & x \in \mathfrak{X} \end{array} \right), \xi \in \mathcal{U}.$$

The elements of $\mathcal{U}$ are called *scenarios*. It is not known which scenario $\xi \in \mathcal{U}$ will occur making the above problem an uncertain multiobjective optimization problem. For any fixed choice of $\xi \in \mathcal{U}$ we have a deterministic multiobjective optimization problem

$$(P(\xi)) \qquad \left(\begin{array}{ll} \min & f(x,\xi) := \begin{pmatrix} f_1(x,\xi) \\ f_2(x,\xi) \\ \vdots \\ f_k(x,\xi) \end{pmatrix} \\ \text{s.t.} & x \in \mathfrak{X} \end{array} \right)$$

for which optimal solutions are defined in the sense of (Pareto-)efficiency:

Notation 1.

- *Let $y, y' \in \mathbb{R}^k$. In what follows, notation $y' \leq y$ and $y' < y$ are both meant componentwise, i.e., $y'_i \leq y_i$ and $y'_i < y_i$, respectively, for all $i = 1, \dots, k$. We say that y'* dominates *y if $y' \leq y$ and there exists some $i \in \{1, \dots, l\}$ with $y'_i < y_i$.*
- *Let $x, x' \in \mathfrak{X}$. We say that x' dominates x if $f(x')$ dominates $f(x)$. A solution x is called efficient, if there does not exist a solution x' which dominates x.*

The *domination property* for the deterministic multiobjective optimization problem $(P(\xi))$ says: For every $x \in \mathfrak{X}$, either x is efficient or there exists an efficient solution $x' \in \mathfrak{X}$ which dominates x. It is known (e.g., in [24]) that the domination property holds for $(P(\xi))$ if $\mathfrak{X}$ is finite and if $\mathfrak{X}$ is compact and the objective functions $f_i(\cdot, \xi)$ are continuous in x for all $i = 1, \dots, k$.

For the results in this paper, let us assume that one of the following conditions is satisfied':

- $\mathfrak{X}$ and $\mathcal{U}$ are both finite sets,
- $\mathcal{U}$ is finite, $\mathfrak{X}$ is compact and $f(\cdot, \xi) : \mathfrak{X} \to \mathbb{R}^k$ is continuous in x for every fixed $\xi \in \mathcal{U}$,
- $\mathfrak{X}$ is finite, $\mathcal{U}$ is compact and $f(x, \cdot) : \mathcal{U} \to \mathbb{R}^k$ is continuous in ξ for every fixed $x \in \mathfrak{X}$,
- $\mathcal{U}$ and $\mathfrak{X}$ are both compact and $f : \mathfrak{X} \times \mathcal{U} \to \mathbb{R}^k$ is jointly continuous in (x, ξ).

Each of these assumptions guarantees that all minima and maxima exist, i.e., that

- $(P(\xi))$ has the domination property for all fixed $\xi \in \mathcal{U}$.
- $\max_{\xi \in \mathcal{U}} f(x, \xi)$ exists for every fixed $x \in \mathfrak{X}$,

The assumptions hold in many problems studied in the literature and in some applications.

For the uncertain problem, several concepts on how to define *robust efficiency* have been proposed. Here, we consider minmax robust efficiency and lightly robust efficiency. Our goal is to compare minmax robust efficient and lightly robust efficient solutions to the solutions we would obtain without considering robustness, i.e., the *nominal efficient solutions*.

2.1 Nominal efficiency

As usual in robust optimization (e.g., in [5] and many other references) we assume that a *nominal scenario* $\hat{\xi} \in \mathcal{U}$ is known. This is the standard scenario one would usually take if robustness issues do not play a role. It might be the undisturbed or the most likely scenario, or it contains the parameters which have been measured without any deviation. We define $f^{\text{nom}}(x) := f(x, \hat{\xi})$ and the *nominal problem* as

$$(P^{\text{nom}}) \qquad \left(\begin{array}{ll} \min & f^{\text{nom}}(x) = \begin{pmatrix} f_1(x,\hat{\xi}) \\ f_2(x,\hat{\xi}) \\ \vdots \\ f_k(x,\hat{\xi}) \end{pmatrix} \\ \text{s.t.} & x \in \mathfrak{X} \end{array} \right).$$

Note that (P^{nom}) is a deterministic multiobjective optimization problem. It is the problem which is 'usually' solved, i.e., when no robustness is taken into account.

Definition 2. *We denote the set of efficient solutions to* (P^{nom}) *by* X^{nom}. *Solutions* $x \in X^{\text{nom}}$ *are called* nominal efficient. *For* $x \in \mathfrak{X}$, *we furthermore call* $f^{\text{nom}}(x)$ *its* nominal quality.

Since (P^{nom}) equals $(P(\hat{\xi}))$, it has the domination property.

Lemma 3. *For every* $x \in \mathfrak{X}$ *there exists* $x' \in \mathfrak{X}^{\text{nom}}$ *with* $f^{\text{nom}}(x') \leq f^{\text{nom}}(x)$.

2.2 Minmax robust efficiency

Minmax robustness is the most widely used concept in single-objective robust optimization (see a summary in [1]). Several generalizations to the multiobjective case have been proposed. Here we use the concept of *point-based minmax robustness* as proposed in [12, 21]. In case of objective-wise uncertainty (called *owu* in [10]), i.e., if the uncertainty is independent between the objective functions, point-based robustness coincides with set-based robustness [10] and hull-based robustness [7].

In order to find minmax robust efficient solutions, we define the worst case objective function f^{wc} as

$$f^{\text{wc}}(x) := \begin{pmatrix} \max_{\xi \in \mathcal{U}} f_1(x,\xi) \\ \max_{\xi \in \mathcal{U}} f_2(x,\xi) \\ \vdots \\ \max_{\xi \in \mathcal{U}} f_k(x,\xi) \end{pmatrix}.$$

The resulting optimization problem in the worst case is given as

$$(P^{\text{wc}}) \qquad \left(\begin{array}{ll} \min & f^{\text{wc}}(x) = \begin{pmatrix} \max_{\xi \in \mathcal{U}} f_1(x,\xi) \\ \max_{\xi \in \mathcal{U}} f_2(x,\xi) \\ \vdots \\ \max_{\xi \in \mathcal{U}} f_k(x,\xi) \end{pmatrix} \\ \text{s.t.} & x \in \mathfrak{X} \end{array} \right).$$

The problem (P^{wc}) is again a deterministic multiobjective optimization problem.

Definition 4. *Let X^{wc} be the set of efficient solutions to (P^{wc}). Solutions $x \in X^{\text{wc}}$ are called* minmax robust efficient. *For $x \in \mathfrak{X}$, we furthermore call $f^{\text{wc}}(x)$ its* worst case objective value.

From our general assumptions we can conclude that (P^{wc}) has the domination property.

Lemma 5. *For every $x \in \mathfrak{X}$ there exists $x' \in \mathfrak{X}^{\text{wc}}$ with $f^{\text{wc}}(x') \leq f^{\text{wc}}(x)$.*

Proof. If $\mathfrak{X}$ is finite, it is trivial that the Lemma holds. Otherwise, $\mathfrak{X}$ is compact and we have to distinguish two cases:

- Either $\mathcal{U}$ is finite and f is continuous in x for every fixed ξ. Then, f^{wc} is continuous as maximum of a finite set of continuous functions.
- Or both, $\mathfrak{X}$ and $\mathcal{U}$ are compact and f is jointly continuous in x and ξ. Then f^{wc} is continuous due to Berge's theorem, see, e.g., [4].

□

In case of objective-wise uncertainty there exists a worst case scenario $\bar{\xi} \in \mathcal{U}$ for which all objective functions simultaneously take their maxima. In this case, (P^{wc}) equals $(P(\bar{\xi}))$ and is hence a deterministic problem. This need not hold if the same uncertain parameter influences more than one of the objective functions.

2.3 Lightly robust efficiency

Finally, we introduce the concept of lightly robust efficient solutions. The idea comes from light robustness in single-objective optimization (see [11, 25]) and focuses on finding solutions which are not too bad in the nominal case. Light robustness was generalized in [18, 20] to the multiobjective case as follows: one first determines the set of efficient solutions X^{nom} for the nominal scenario $\hat{\xi}$. We allow a lightly robust efficient solution to be a bit worse than the efficient solutions in the nominal scenario. The deviation from the objective values in the nominal scenario should be bounded by some given $\varepsilon \in \mathbb{R}^k$, where ε_i bounds the deviation in objective function f_i^{nom}. In order to ensure this, we define for each $\hat{x} \in X^{\text{nom}}$

$$(P^{\text{light},\varepsilon}(\hat{x})) \qquad \left(\begin{array}{ll} \min & f^{\text{wc}}(x) = \begin{pmatrix} \max_{\xi\in\mathcal{U}} f_1(x,\xi) \\ \max_{\xi\in\mathcal{U}} f_2(x,\xi) \\ \vdots \\ \max_{\xi\in\mathcal{U}} f_k(x,\xi) \end{pmatrix} \\ \text{s.t.} & f^{\text{nom}}(x) \leq f^{\text{nom}}(\hat{x}) + \varepsilon \\ & x \in \mathfrak{X} \end{array} \right),$$

i.e., among all solutions which are only a bit worse than $\hat{x}$ in the nominal scenario we take the ones which are efficient in the worst case, i.e., which are minmax robust efficient within the set

$$F^{\text{light},\varepsilon}(\hat{x}) := \{x \in \mathfrak{X} : f^{\text{nom}}(x) \leq f^{\text{nom}}(\hat{x}) + \varepsilon\}$$

of feasible solutions to $(P^{\text{light},\varepsilon}(\hat{x}))$.

Definition 6. *For $\hat{x} \in \mathfrak{X}^{\text{nom}}$, let $X^{\text{light},\varepsilon}(\hat{x})$ be the set of efficient solutions to $(P^{\text{light},\varepsilon}(\hat{x}))$. Solutions $x \in X^{\text{light},\varepsilon} := \bigcup_{\hat{x}\in\mathfrak{X}^{\text{nom}}} X^{\text{light},\varepsilon}(\hat{x})$ are called* lightly robust efficient.

Due to its closedness, the feasible set $F^{\text{light},\varepsilon}(\hat{x})$ is compact if $\mathfrak{X}$ is compact, and finite if $\mathfrak{X}$ is finite. With the same reasoning as for (P^{wc}) we hence conclude that the domination property holds for $(P^{\text{light},\varepsilon}(\hat{x}))$.

Lemma 7. *For every $x \in F^{\text{light},\varepsilon}(\hat{x})$ there exists $x' \in X^{\text{light},\varepsilon}(\hat{x})$ with $f^{\text{wc}}(x') \leq f^{\text{wc}}(x)$.*

3 Comparing sets of robust efficient solutions

In (non-robust) multiobjective optimization, the quality of a set of solutions $X \subseteq \mathfrak{X}$ is usually evaluated by looking at the images of the solutions $f(X)$ in the objective space. If X is the set of efficient solutions, their images $f(X)$ are called the *efficient front*. In order to compare the sets of nominal efficient solutions X^{nom}, of minmax robust efficient solutions X^{wc}, and of lightly robust efficient solutions $X^{\text{light},\varepsilon}$, we proceed similarly: we look at the images under the objective function f. However, the objective function values not only depend on $x \in \mathfrak{X}$ but also on the scenario which occurs; we hence get different objective function values for each scenario $\xi \in \mathcal{U}$ and an efficient point in the nominal scenario need not be an efficient point in other scenarios. To consider properties of a set $X \subseteq \mathfrak{X}$ of solutions (specifically for $X = X^{\text{nom}}, X = X^{\text{wc}}$, or $X = X^{\text{light},\varepsilon}$) we propose to evaluate, i.e., to calculate the objective function values of, X in two cases:

- A first evaluation should consider the nominal case, i.e., discuss

$$f^{\text{nom}}(X) = \{f(x, \hat{\xi}) : x \in X\}.$$

  From a practical point of view such an evaluation makes sense since it shows what to expect in the most likely (or undisturbed) scenario. Clearly, $f^{\text{nom}}(X^{\text{nom}})$ shows the efficient front of the problem (P^{nom}).

- The second evaluation takes a robust perspective considering X under its worst case, i.e., we evaluate

$$f^{\text{wc}}(X) := \{f^{\text{wc}}(x) : x \in X\} = \left\{ \begin{pmatrix} \max_{\xi \in \mathcal{U}} f_1(x, \xi) \\ \vdots \\ \max_{\xi \in \mathcal{U}} f_k(x, \xi) \end{pmatrix} : x \in X \right\}.$$

Note that in contrast to $f^{\text{nom}}(X)$ which is always evaluated under the scenario $\hat{\xi} \in \mathcal{U}$, the scenarios which are relevant for evaluating $f^{\text{wc}}(x)$ depend on the objective function $f_i, i = 1, \ldots, k$ and on the point $x \in X$ itself.

The intuition is that under the nominal scenario, the set $f^{\text{nom}}(X^{\text{nom}})$ is better than the set of minmax robust efficient solutions $f^{\text{nom}}(X^{\text{wc}})$ while this result of comparison changes if we evaluate under the worst case objective function f^{wc}, i.e., $f^{\text{wc}}(X^{\text{wc}})$ is better than $f^{\text{wc}}(X^{\text{nom}})$. The set of lightly robust efficient solutions is expected to lie somewhere in between as they are claimed in [18, 20] to be a good compromise between nominal quality and robustness.

To formalize the intuitions described above of robust efficient solutions, we use a set-based order to compare two sets $Y_1, Y_2 \subseteq \mathbb{R}^k$:

Notation 8.

$$Y_1 \prec^{upp} Y_2 \text{ if for all } y \in Y_2 \text{ there exists } y' \in Y_1 \text{ with } y' \leq y$$

$$Y_1 \prec^{low} Y_2 \text{ if for all } y \in Y_1 \text{ there exists } y' \in Y_2 \text{ with } y \leq y'.$$

Denoting $\mathbb{R}^k_{\geq} = \{y \in R^k : y_i \geq 0 \text{ for all } i = 1, \ldots, k\}$ as the nonnegative ordering cone, $Y_1 \prec^{upp} Y_2$ can equivalently be written as

$$Y_1 \prec^{upp} Y_2 \text{ if } Y_1 + \mathbb{R}^k_{\geq} \supseteq Y_2$$

which is known as the *upper set less order*, see [19], and $Y_1 \prec^{low} Y_2$ can equivalently be written as

$$Y_1 \prec^{low} Y_2 \text{ if } Y_2 - \mathbb{R}^k_{\geq} \supseteq Y_1$$

which is known as the *lower set less order*, see again [19].

We first show that evaluating solutions in the worst case always gives a more pessimistic point of view than evaluating in the nominal case, no matter what we choose as the set $X \subseteq \mathfrak{X}$.

Lemma 9. *For every set $X \subseteq \mathfrak{X}$ we have:*

(i) $f^{\text{nom}}(X) \prec^{upp} f^{\text{wc}}(X)$.

(ii) $f^{\text{nom}}(X) \prec^{low} f^{\text{wc}}(X)$.

Proof.

(i) : Let $y \in f^{\text{wc}}(X)$, i.e., $y = f^{\text{wc}}(x, \xi)$ for some $x \in X$ and some $\xi \in \mathcal{U}$. Define $y' := f^{\text{nom}}(x)$. Then $y' \in f^{\text{nom}}(X)$ and for each component $i = 1, \ldots, k$ we have

$$y'_i = f_i(x, \hat{\xi}) \leq \max_{\xi \in \mathcal{U}} f_i(x, \xi) = y_i,$$

hence $y' \leq y$.

(ii) : Let $y \in f^{\text{nom}}(X)$, i.e., $y = f^{\text{nom}}(x, \xi)$ for some $x \in X$ and some $\xi \in \mathcal{U}$. Define $y' := f^{\text{wc}}(x)$. Then $y' \in f^{\text{wc}}(X)$ and analogously to part (i) it follows $y \leq y'$.

□

It is more interesting to compare the different sets of robust efficient points with each other. We start by showing that under the upper set less order relation, X^{nom} is better than any other set $X \subseteq \mathfrak{X}$ in the nominal scenario, and X^{wc} is better than any other set $X \subseteq \mathfrak{X}$ under worst case evaluation.

Lemma 10. *For every set $X \subseteq \mathfrak{X}$ we have:*

(i) $f^{\text{nom}}(X^{\text{nom}}) \prec^{upp} f^{\text{nom}}(X)$.

(ii) $f^{\text{wc}}(X^{\text{wc}}) \prec^{upp} f^{\text{wc}}(X)$.

Proof.

(i) Let $y \in f^{\text{nom}}(X)$, i.e., $y = f^{\text{nom}}(x)$ for some $x \in X$. Due to the domination property for (P^{nom}) (Lemma 3), there exists $x' \in X^{\text{nom}}$ with $f^{\text{nom}}(x') \leq f^{\text{nom}}(x)$. Setting $y' := f^{\text{nom}}(x')$ shows the assertion.

(ii) Now let $y \in f^{\text{wc}}(X)$, i.e., $y = f^{\text{wc}}(x)$ for some $x \in X$. Due to the domination property for (P^{wc}) (Lemma 5) there exists $x' \in X^{\text{wc}}$ with $y' := f^{\text{wc}}(x') \leq f^{\text{wc}}(x) = y$, and the proof is complete.

□

Note that Lemma 10 does *not* hold under the lower set less order relation, not even if we only compare the sets X^{nom} and X^{wc} with each other in the nominal scenario. This is shown in the following small example. Note that the problem in this example only has $k = 2$ objective functions which are objective-wise independent (see [10] for a formal definition); so the relation does not even hold under this rather special condition.

Example 1. *Let two scenarios $\mathcal{U} = \{\hat{\xi}, \bar{\xi}\}$ be given, consider a feasible set $\mathfrak{X}$ which contains only two elements $\mathfrak{X} = \{x_1, x_2\}$ and two objective functions. Let*

$$f(x_1, \hat{\xi}) = \begin{pmatrix} 2 \\ 1 \end{pmatrix}, \qquad f(x_2, \hat{\xi}) = \begin{pmatrix} 0 \\ 3 \end{pmatrix}$$

$$f(x_1, \bar{\xi}) = \begin{pmatrix} 5 \\ 5 \end{pmatrix}, \qquad f(x_2, \bar{\xi}) = \begin{pmatrix} 1 \\ 4 \end{pmatrix},$$

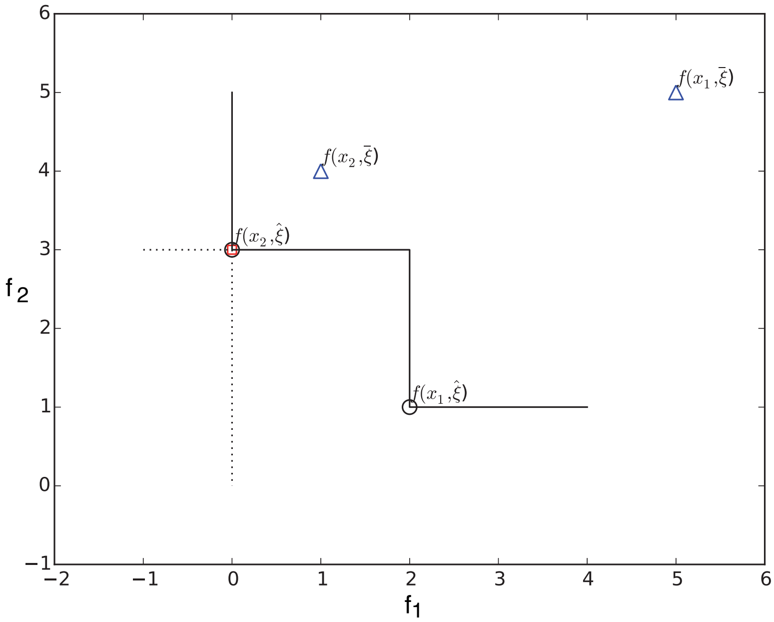


Figure 1: Illustration of Example 1. We compare $f^{\text{nom}}(X^{\text{nom}}) = \{f(x_1,\hat{\xi}), f(x_2,\hat{\xi})\}$ and $f^{\text{nom}}(X^{\text{wc}}) = \{f(x_2,\hat{\xi})\}$.

see Figure 1 for an illustration. In this example, we receive

$$X^{\text{nom}} = \{x_1, x_2\}$$

since their objective function values in the nominal scenario do not dominate each other. For f^{wc} *we receive*

$$f^{\text{wc}}(x_1) = \begin{pmatrix} \max\{2,5\} \\ \max\{1,5\} \end{pmatrix} = \begin{pmatrix} 5 \\ 5 \end{pmatrix}$$
$$f^{\text{wc}}(x_2) = \begin{pmatrix} \max\{0,1\} \\ \max\{3,4\} \end{pmatrix} = \begin{pmatrix} 1 \\ 4 \end{pmatrix},$$

i.e.,

$$X^{\text{wc}} = \{x_2\}.$$

Hence,

$$f^{\text{nom}}(X^{\text{nom}}) = \left\{ \begin{pmatrix} 2 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 3 \end{pmatrix} \right\} \quad \textit{and} \quad f^{\text{nom}}(X^{\text{wc}}) = \left\{ \begin{pmatrix} 0 \\ 3 \end{pmatrix} \right\}.$$

As Lemma 10 says, we have

$$f^{\text{nom}}(X^{\text{nom}}) \prec^{upp} f^{\text{nom}}(X^{\text{wc}}),$$

but for the lower set less order this does not hold. We even receive

$$f^{\text{nom}}(X^{\text{wc}}) \prec^{low} f^{\text{nom}}(X^{\text{nom}}).$$

We continue with analyzing the set of lightly robust efficient solutions $X^{\text{light},\varepsilon}$. We start with a simple observation which follows directly from Definition 6.

Lemma 11. *For every $\varepsilon \geq 0$ we have*

(i) $f^{\mathrm{nom}}(X^{\mathrm{light},\varepsilon}) \prec^{low} f^{\mathrm{nom}}(X^{\mathrm{nom}}) + \{\varepsilon\}$,

(ii) $f^{\mathrm{nom}}(X^{\mathrm{light},\varepsilon}) \prec^{upp} f^{\mathrm{nom}}(X^{\mathrm{nom}}) + \{\varepsilon\}$.

Proof.

(i) Let $y \in f^{\mathrm{nom}}(X^{\mathrm{light},\varepsilon})$, i.e., $y = f^{\mathrm{nom}}(x)$ for $x \in X^{\mathrm{light},\varepsilon}$. Then there exists a solution $\hat{x} \in X^{\mathrm{nom}}$ such that x is an efficient solution to $(P^{\mathrm{light},\varepsilon}(\hat{x}))$. In particular,

$$f^{\mathrm{nom}}(x) \leq f^{\mathrm{nom}}(\hat{x}) + \varepsilon.$$

We define $y' := f^{\mathrm{nom}}(\hat{x}) + \varepsilon \in f^{\mathrm{nom}}(X^{\mathrm{nom}}) + \{\varepsilon\}$ and receive that $y = f^{\mathrm{nom}}(x) \leq y'$.

(ii) Let $y \in f^{\mathrm{nom}}(X^{\mathrm{nom}}) + \{\varepsilon\}$, i.e., $y = f^{\mathrm{nom}}(\hat{x}) + \varepsilon$ for some $\hat{x} \in X^{\mathrm{nom}}$. Due to the domination property for $(P^{\mathrm{light},\varepsilon}(\hat{x}))$ (Lemma 7) there exists $x' \in X^{\mathrm{light},\varepsilon}(\hat{x})$. In particular,

$$f^{\mathrm{nom}}(x') \leq f^{\mathrm{nom}}(\hat{x}) + \varepsilon.$$

With $y' := f^{\mathrm{nom}}(x')$ we hence receive $y' = f^{\mathrm{nom}}(x') \leq f^{\mathrm{nom}}(\hat{x}) + \varepsilon = y$.

□

Together with Lemma 10 we summarize

$$f^{\mathrm{nom}}(X^{\mathrm{nom}}) \prec^{upp} f^{\mathrm{nom}}(X^{\mathrm{light},\varepsilon}) \prec^{upp} f^{\mathrm{nom}}(X^{\mathrm{nom}}) + \{\varepsilon\},$$

i.e., for small $\varepsilon > 0$ the evaluation of X^{nom} and $X^{\mathrm{light},\varepsilon}$ under the nominal scenario differs only slightly. The next lemma analyzes what might happen to lightly robust efficient solutions in the worst case.

Lemma 12.

(i) $f^{\mathrm{wc}}(X^{\mathrm{light},\varepsilon}) \prec^{upp} f^{\mathrm{wc}}(X^{\mathrm{nom}})$ *for all* $\varepsilon \geq 0$ *and*

(ii) $f^{\mathrm{wc}}(X^{\mathrm{light},\varepsilon_2}) \prec^{upp} f^{\mathrm{wc}}(X^{\mathrm{light},\varepsilon_1})$ *for all* $0 \leq \varepsilon_1 \leq \varepsilon_2$.

Proof.

(i) Let $y = f^{\mathrm{wc}}(x)$ for $x \in X^{\mathrm{nom}}$. Then $x \in F^{\mathrm{light},\varepsilon}(x)$, i.e., it is feasible for $(P^{\mathrm{light},\varepsilon}(x))$. Due to the domination property for $(P^{\mathrm{light},\varepsilon}(x))$ (Lemma 7) there exists $x' \in X^{\mathrm{light},\varepsilon}(x)$ with $f^{\mathrm{wc}}(x') \leq f^{\mathrm{wc}}(x)$. We hence have $y' := f^{\mathrm{wc}}(x') \leq f^{\mathrm{wc}}(x) = y$.

(ii) Now let $y = f^{\mathrm{wc}}(x)$ for $x \in X^{\mathrm{light},\varepsilon_1}$. Then there exists $\hat{x} \in X^{\mathrm{nom}}$ such that $x \in F^{\mathrm{light},\varepsilon_1}(\hat{x})$. Since $\varepsilon_2 \geq \varepsilon_1$ we know that $x \in F^{\mathrm{light},\varepsilon_2}(\hat{x})$. We again use the domination property (Lemma 7) for $(P^{\mathrm{light},\varepsilon_2}(\hat{x}))$ and receive $x' \in X^{\mathrm{light},\varepsilon_2}$ which satisfies $y' := f^{\mathrm{wc}}(x') \leq f^{\mathrm{wc}}(x) = y$.

□

Note that the statements of Lemma 12 do not hold for the lower set less order. They also cannot be mirrored to the nominal case, i.e., it is *not* true in general that

1. $f^{\mathrm{nom}}(X^{\mathrm{light},\varepsilon}) \prec f^{\mathrm{nom}}(X^{\mathrm{wc}})$ for $\varepsilon > 0$, and that
2. $f^{\mathrm{nom}}(X^{\mathrm{light},\varepsilon_1}) \prec f^{\mathrm{nom}}(X^{\mathrm{light},\varepsilon_2})$ for $0 \leq \varepsilon_1 < \varepsilon_2$,

neither for $\prec$ being the upper set less order nor for the lower set less order. This is illustrated next.

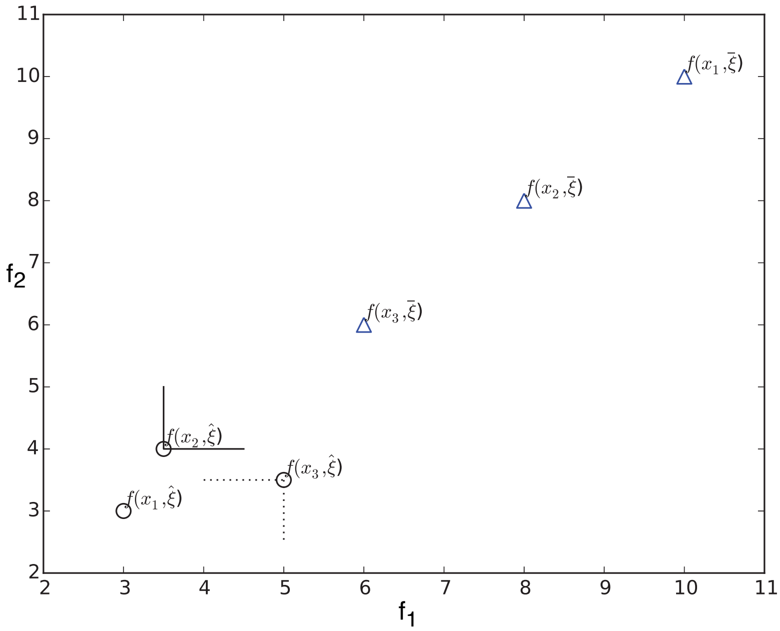


Figure 2: Illustration of Example 2. Lemma 12 neither holds for $\prec^{low}$ nor can it be mirrored to the nominal case for $\prec^{upp}$.

Example 2. *Let two scenarios* $\mathcal{U} = \{\hat{\xi}, \bar{\xi}\}$ *be given, consider a feasible set* $\mathfrak{X} = \{x_1, x_2, x_3\}$ *and two objective functions. Let*

$$f(x_1, \hat{\xi}) = \begin{pmatrix} 3 \\ 3 \end{pmatrix}, \quad f(x_2, \hat{\xi}) = \begin{pmatrix} 3.5 \\ 4 \end{pmatrix}, \quad f(x_3, \hat{\xi}) = \begin{pmatrix} 5 \\ 3.5 \end{pmatrix},$$

$$f(x_1, \bar{\xi}) = \begin{pmatrix} 10 \\ 10 \end{pmatrix}, \quad f(x_2, \bar{\xi}) = \begin{pmatrix} 8 \\ 8 \end{pmatrix}, \quad f(x_3, \bar{\xi}) = \begin{pmatrix} 6 \\ 6 \end{pmatrix}.$$

Then

$$X^{\text{nom}} = \{x_1\},\ X^{\text{light},1} = \{x_2\},\ X^{\text{light},2} = \{x_3\},\ X^{\text{wc}} = \{x_3\}$$

This example is illustrated in Figure 2.

The next lemma analyzes what happens in the nominal scenario when lightly robust efficient solutions and minmax efficient solutions are compared.

Lemma 13. *Let* $\varepsilon \geq 0$ *and* $x \in X^{\text{light},\varepsilon}$. *Then there does not exist* $x' \in X^{\text{wc}}$ *which is at least as good as* x *with respect to* f^{nom} *and dominates* x *with respect to* f^{wc}.

Proof. Let $x' \in X^{\text{wc}}$ and $x \in X^{\text{light},\varepsilon}$. Assume that $f^{\text{nom}}(x') \leq f^{\text{nom}}(x)$. We know that $f^{\text{nom}}(x) \leq f^{\text{nom}}(\hat{x}) + \epsilon$ for some $\hat{x} \in X^{\text{nom}}$. So $f^{\text{nom}}(x') \leq f^{\text{nom}}(\hat{x}) + \epsilon$ holds. Hence, $x' \in F^{\text{light},\varepsilon}(\hat{x})$ and due to $x' \in X^{wc}$ we conclude that $x' \in \mathfrak{X}^{\text{light},\varepsilon}(\hat{x})$, i.e., both x and x' are lightly robust efficient and consequently, do not dominate each other under f^{wc}. Thus, the lemma holds. □

What we have learnt about lightly robust efficient solutions. We summarize our main findings with respect to lightly robust efficient solutions: First, the set of lightly robust efficient solutions lies (in the worst case) always between the set of nominal efficient and the set of worst case efficient solutions (see Lemma 10 and Lemma 12):

$$f^{\mathrm{wc}}(X^{\mathrm{wc}}) \prec^{upp} f^{\mathrm{wc}}(X^{\mathrm{light},\varepsilon_2}) \prec^{upp} f^{\mathrm{wc}}(X^{\mathrm{light},\varepsilon_1}) \prec^{upp} f^{\mathrm{wc}}(X^{\mathrm{nom}}) \tag{1}$$

for $\varepsilon_1 \leq \varepsilon_2$. Hence, choosing lightly robust efficient solutions might be a good compromise between nominal and minmax efficient solutions. Second, the larger ε is chosen, the more robustness we gain.

For the nominal scenario, due to Lemma 10 and Lemma 11 we furthermore know that

$$f^{\mathrm{nom}}(X^{\mathrm{nom}}) \prec^{upp} f^{\mathrm{nom}}(X^{\mathrm{light},\varepsilon}) \prec f^{\mathrm{nom}}(X^{\mathrm{nom}}) + \{\varepsilon\} \tag{2}$$

where the second relation holds for both, the upper set less order $\prec^{upp}$ and the lower set less order $\prec^{low}$, which means that in the nominal scenario, the set of lightly robust efficient solutions gets more similar to the set of nominal efficient solutions if ε is decreased.

4 Examples and Illustration

So far, we have analyzed the three different sets of nominal, lightly robust efficient and minmax robust efficient solutions in the nominal case and in the worst case. In this section, we illustrate our findings with some examples.

We first look at a bi-objective optimization problem given as follows.

Example 3.

$$\left(\begin{array}{lll} \min & f_1(x,\xi_1) = x_1^2 + \xi_1 x_2^2, \\ & f_2(x,\xi_2) = (\xi_2 x_1 - 5)^2 + (x_2 - 5)^2 \\ s.t. & 0 \leq x_1 \leq 4 \\ & 0 \leq x_2 \leq 3, \end{array}\right)_{\xi_1 \in \mathcal{U}_1, \xi_2 \in \mathcal{U}_2.} \tag{3}$$

where $\mathcal{U}_1 = \{-6, -1, 0.5, 1\}$ *and* $\mathcal{U}_2 = \{-2, -1, 1, 2\}$ *and the nominal values for the uncertain parameters are* $\hat{\xi_1} = -1$ *and* $\hat{\xi_2} = 1$*. This problem is a variation of the Binh and Korn function [6], which minimizes two quadratic functions within the given ranges of decision variables. In this problem,* ξ_1 *and* ξ_2 *are independent from each other and there exists a single worst case scenario* $\xi_1 = 1$*,* $\xi_2 = -2$*.*

Figure 3 shows X^{wc}*,* $X^{\mathrm{light},15}$*,* $X^{\mathrm{light},10}$*, and* X^{nom} *evaluated in the nominal case. We can see the relationship*

$$f^{\mathrm{nom}}(X^{\mathrm{nom}}) \prec^{upp} f^{\mathrm{nom}}(X^{\mathrm{light},10}) \prec^{upp} f^{\mathrm{nom}}(X^{\mathrm{light},15}) \prec^{low/upp} f^{\mathrm{nom}}(X^{\mathrm{nom}}) + 15.$$

Figure 4 illustrates X^{wc}*,* $X^{\mathrm{light},15}$*,* $X^{\mathrm{light},10}$*, and* X^{nom} *evaluated in the worst case. In the figure, we observe the results of Lemma 10 and Lemma 12, namely,*

$$f^{\mathrm{wc}}(X^{\mathrm{wc}}) \prec^{upp} f^{\mathrm{wc}}(X^{\mathrm{light},15}) \prec^{upp} f^{\mathrm{wc}}(X^{\mathrm{light},10}) \prec^{upp} f^{\mathrm{wc}}(X^{\mathrm{nom}}).$$

In this specific problem, we have

$$f^{\mathrm{nom}}(X^{\mathrm{light},10}) \prec^{upp} f^{\mathrm{nom}}(X^{\mathrm{light},15}) \prec^{upp} f^{\mathrm{nom}}(X^{wc})$$

even though it does not hold in general. The example also illustrates that $f^{\mathrm{nom}}(X^{\mathrm{nom}}) \prec^{low} f^{\mathrm{nom}}(X^{\mathrm{light},\varepsilon})$ *and* $f^{\mathrm{nom}}(X^{\mathrm{nom}}) \prec^{low} f^{\mathrm{nom}}(X^{\mathrm{wc}})$ *need not hold.*

We also observe that a solution in X^{nom} *can have very bad objective function values in the worst case compared to lightly and minmax robust efficient solutions. On the other hand, gaining minmax robust efficiency comes at a high price: there has to be a great sacrifice on the nominal quality of the solutions as the minmax robust efficient solutions are very far from the nominal solutions when evaluated in the nominal case. In this example, lightly robust solutions are a good compromise.*

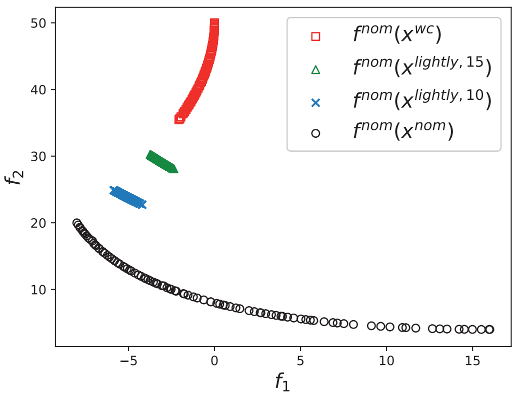


Figure 3: Evaluation in the nominal case of Example 3

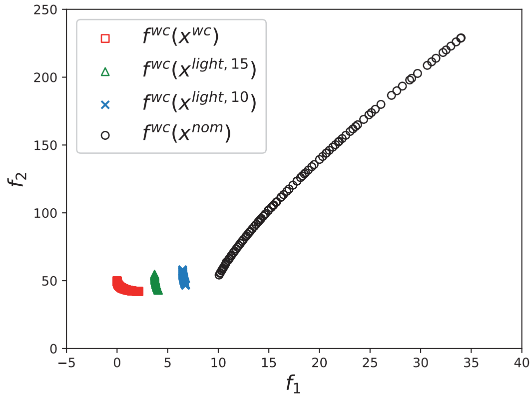


Figure 4: Evaluation in the worst case of Example 3

The example also illustrates a general property, namely, which of the depicted sets form "efficient fronts", i.e., in which cases the depicted points do not dominate each other. This holds per definition for $f^{\text{nom}}(X^{\text{nom}})$, $f^{\text{wc}}(X^{\text{wc}})$, and $f^{\text{wc}}(X^{\text{light},\varepsilon})$, i.e., we obtain a nominal efficient front for the nominal robust efficient points and a minmax robust efficient front for the minmax robust efficient and for the different lightly robust efficient points. In contrast to this, Figures 4 shows that the points in $f^{\text{wc}}(X^{\text{nom}})$ may dominate each other.

Next, we consider a more interesting example where the worst case depends on the solution x.

Example 4.

$$\begin{pmatrix} \min & f_1(x,\xi_1) = \xi_1 x_1 + x_2, \\ & f_2(x,\xi_2) = -x_1 - \xi_2 x_2 \\ s.t. & -2 \le x_1 \le 2 \\ & -2 \le x_2 \le 2, \end{pmatrix}_{\xi=(\xi_1,\xi_2)^T \in \mathcal{U}.} \tag{4}$$

The uncertainty set $\mathcal{U}$ is:

$$\mathcal{U} = \left\{ \begin{pmatrix} -3 \\ 1.5 \end{pmatrix}, \begin{pmatrix} -1 \\ 2 \end{pmatrix}, \begin{pmatrix} 1 \\ 2.5 \end{pmatrix} \right\}.$$

The nominal values for the uncertain parameters are $\hat{\xi}_1 = -1$ and $\hat{\xi}_2 = 2$. Finding a worst case for some given and feasible x in this example can be written as the following two-objective optimization problem

$$\begin{pmatrix} \max & f_1(x,\xi_1) = \xi_1 x_1 + x_2, \\ & f_2(x,\xi_2) = -x_1 - \xi_2 x_2 \\ s.t. & \xi \in \mathcal{U} \end{pmatrix}. \tag{5}$$

We see that the worst case depends on the solution x: for $-2 \le x_1 \le 0$ and $0 \le x_2 \le 2$, the worst case is $\xi = (-3, 1.5)^T$. For $-2 \le x_1 \le 0$ and $-2 \le x_2 \le 0$, there does not exist a single worst case. We observe that no pair of the three scenarios in $\mathcal{U}$ dominates each other, hence the set of non-dominated solutions of the maximization problem (5) is $\mathcal{U}$ itself. Similarly, for $0 \le x_1 \le 2, -2 \le x_2 \le 0$, the worst case is $\xi = (1, 2.5)^T$ and for $0 \le x_1 \le 2$ and $0 \le x_2 \le 2$, the set of worst-case scenarios is again $\mathcal{U}$. Since we use point-based minmax robust efficiency, we need not worry about the existence of

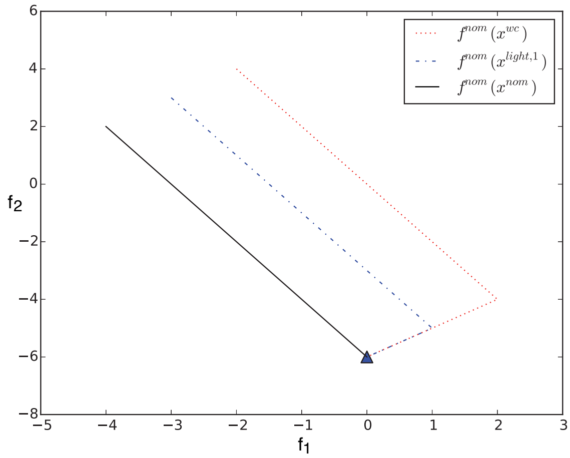


Figure 5: Evaluation in the nominal case of Example 4

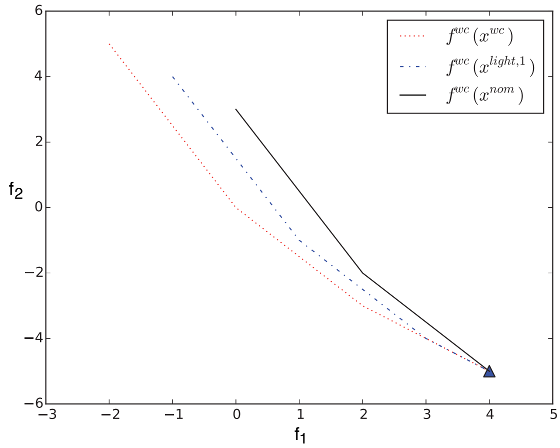


Figure 6: Evaluation in the worst case of Example 4

multiple worst-case scenarios, but compute the worst-case objective function f^{wc} *by taking the componentwise maximum. We hence receive*

$$\begin{aligned} X^{\mathrm{nom}} &= \{(2, x_2) : -2 \leq x_2 \leq 2\}, \\ X^{\mathrm{light},1} &= \{(1, x_2) : -2 \leq x_2 \leq 2\} \cup \{(x_1, 2) : 1 \leq x_1 \leq 2\}, \\ X^{\mathrm{wc}} &= \{(0, x_2) : -2 \leq x_2 \leq 2\} \cup \{(x_1, 2) : 0 \leq x_1 \leq 2\}. \end{aligned}$$

Figure 5 shows $X^{\mathrm{nom}}, X^{\mathrm{light},1}$ *and* X^{wc} *in the nominal case and Figure 6 shows the three sets of solutions in the worst case. As shown in Figure 5, this example is in accordance with our results of Section 3:*

$$f^{\mathrm{nom}}(X^{\mathrm{nom}}) \prec^{upp} f^{\mathrm{nom}}(X^{\mathrm{light},1}) \prec^{upp} f^{\mathrm{nom}}(X^{\mathrm{nom}}) + 1.$$

In this example we also receive $f^{\mathrm{nom}}(X^{\mathrm{light},1}) \prec^{upp} f^{\mathrm{nom}}(X^{wc})$ *although it does not hold for general problems. Figure 6 illustrates our results on the worst case evaluation, namely*

$$f^{\mathrm{wc}}(X^{\mathrm{wc}}) \prec^{upp} f^{\mathrm{nom}}(X^{\mathrm{light},1}) \prec^{upp} f^{wc}(X^{\mathrm{nom}}).$$

The example has a particularity, namely the solution $x = (2, 2)^T$ *is a common element in all three sets of solutions and hence naturally a good choice as a final solution to gain best possible objective function values in both, the nominal and the worst case. The solution is indicated with a triangle in the two figures. We also observe that* $f^{\mathrm{nom}}(X^{\mathrm{nom}})$ *is not that far from the minmax robust efficient front in the worst case while the two sets differ significantly in the nominal case. Hence, in this example, much quality in the nominal case has to be sacrificed to gain minmax robust efficiency, i.e., the price to gain robustness is rather high.*

The observations on the examples motivate us to further analyze the trade-off between the nominal quality and the robustness of the solutions in the next section.

5 Utilizing the price of robustness in decision making

The price of robustness has been popular in single-objective robust optimization since its proposal in [5]. In this section we propose how to measure the price of robustness in

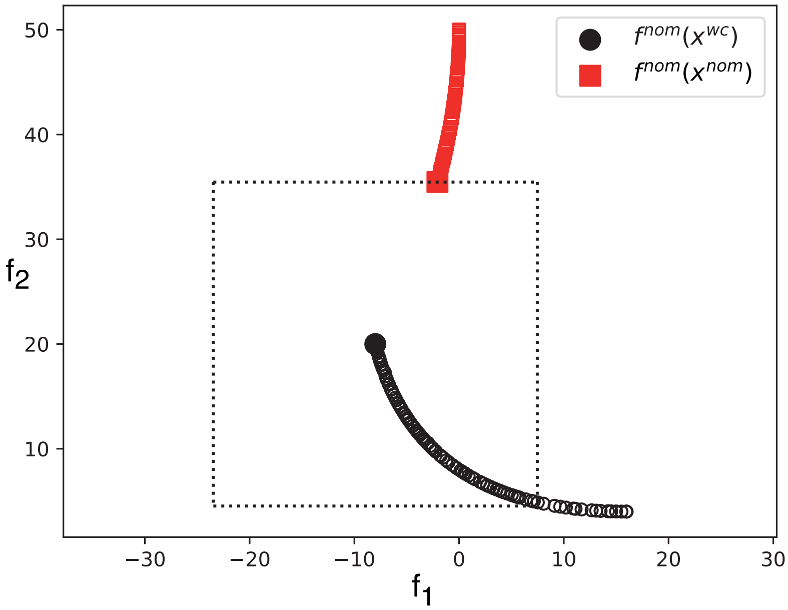


Figure 7: A nominal efficient solution and the closest minmax robust efficient solution.

a multiobjective setting, i.e., how much nominal quality has to be sacrificed in order to receive a minmax robust efficient solution. We first define the price of robustness, and then sketch ideas on how to utilize it to support a decision maker to find a desired solution which is satisfactory in both respects, nominal quality and robustness.

Definition 14. *Let $x \in \mathfrak{X}$ be a feasible solution to $P_{\mathcal{U}}$. We define its* price of robustness *as the objective value of the minimization problem*

$$\text{price}(x) \quad = \quad \inf_{\bar{x} \in X^{\text{wc}}} \|f^{\text{nom}}(x) - f^{\text{nom}}(\bar{x})\|_\infty,$$

where $\|\cdot\|_\infty$ denotes the infinity-metric.

Note that a minimum of the above optimization problem need not always exist, not even under the assumptions we stated in Section 2. However, a minimum exists for linear optimization problems and if the objective function f^{wc} is continuous and strictly quasiconcave [3]. For $x \in X^{\text{nom}}$ being efficient in the nominal case, price(x) tells us how much nominal quality we have to sacrifice in one of the objective functions if we replace x by its (closest) robust efficient solution. Instead of using $\|\cdot\|_\infty$ we could also use another norm, e.g., $\|\cdot\|_1$ would give us the average nominal quality over all objective functions we lose when changing x to a minmax robust efficient solution. Clearly, a points $x \in \mathfrak{X}$ is minmax robust efficient if and only if its price of robustness is zero, i.e.,

$$x \in X^{\text{wc}} \Longleftrightarrow \text{price}(x) = 0.$$

Geometrically, for computing price(x), the closest minmax robust efficient solution to x is chosen from X^{wc} with respect to $\|\cdot\|_\infty$. The situation is illustrated in Figure 7. In the figure, the nominal efficient solution is marked by a bullet and the closest minmax robust efficient solution is marked by a filled square on the robust efficient front. The big square centered in the nominal efficient solution is the visual representation of $\|\cdot\|_\infty$.

The relation to lightly robust efficient solutions is analyzed next. In Lemma 31 in [18] it was shown that there exists $\varepsilon \geq 0$ such that there exists a solution x to $(P^{\text{light},\varepsilon}(\hat{x}))$ which is minmax robust efficient, i.e., with $\text{price}(x) = 0$. We strengthen this result by specifying the size of ε and we extend it to the situation in which all solutions to $(P^{\text{light},\varepsilon}(\hat{x}))$ are minmax robust efficient.

Lemma 15. *Let $\hat{x} \in X^{\text{nom}}$ be given. Then the following hold.*

- $\varepsilon \geq \text{price}(\hat{x})$ *if and only if there is a solution x to $(P^{\text{light},\varepsilon}(\hat{x}))$ with* $\text{price}(x) = 0$, *and*
- $\varepsilon \geq \sup_{\bar{x} \in X^{\text{wc}}} \|f^{\text{nom}}(\bar{x}) - f^{\text{nom}}(\hat{x})\|_\infty$ *if and only if all solutions x to $(P^{\text{light},\varepsilon}(\hat{x}))$ satisfy* $\text{price}(x) = 0$.

Proof.

- Let $\varepsilon \geq \text{price}(\hat{x})$, i.e., there exists $\bar{x} \in X^{\text{wc}}$ with $\|\hat{x} - \bar{x}\|_\infty \leq \varepsilon$. Due to $\bar{x}$ being minmax robust efficient, it is an optimal solution to $(P^{\text{light},\varepsilon}(\hat{x}))$ and satisfies $\text{price}(\bar{x}) = 0$. On the other hand, let x be an optimal solution to $(P^{\text{light},\varepsilon}(\hat{x}))$ with $\text{price}(x) = 0$, i.e., $x \in X^{\text{wc}}$. Since $x \in F^{\text{light},\varepsilon}(\hat{x})$ we furthermore know $\|f_i^{\text{nom}}(x) - f_i^{\text{nom}}(\hat{x})\|_\infty \leq \varepsilon$ for all objectives $i = 1, \ldots, k$. We hence conclude

$$\varepsilon \geq \|f^{\text{nom}}(x) - f^{\text{nom}}(\hat{x})\|_\infty \geq \inf_{x \in X^{\text{wc}}} \|f^{\text{nom}}(x) - f^{\text{nom}}(\hat{x})\|_\infty = \text{price}(\hat{x}).$$

- Now let $\varepsilon \geq \sup_{\bar{x} \in X^{\text{wc}}} \|f^{\text{nom}}(\bar{x}) - f^{\text{nom}}(\hat{x})\|_\infty$, i.e., all $\bar{x} \in X^{\text{wc}}$ satisfy $\|\hat{x} - \bar{x}\|_\infty \leq \varepsilon$, hence $X^{\text{wc}} \subseteq F^{\text{light},\varepsilon}(\hat{x})$ and consequently, every solution to $(P^{\text{light},\varepsilon}(\hat{x}))$ is minmax robust efficient. For the reverse direction, if every minmax robust efficient solution $\bar{x} \in X^{\text{wc}}$ is optimal to $(P^{\text{light},\varepsilon}(\hat{x}))$ we have in particular that $X^{\text{wc}} \subseteq F^{\text{light},\varepsilon}(\hat{x})$. Hence, for all $\bar{x} \in X^{\text{wc}}$ we have

$$\varepsilon \geq \|f^{\text{nom}}(\bar{x}) - f^{\text{nom}}(\hat{x})\|_\infty,$$

i.e.,

$$\varepsilon \geq \sup_{x \in X^{\text{wc}}} \|f^{\text{nom}}(x) - f^{\text{nom}}(\hat{x})\|_\infty.$$

□

In practice it is preferable to choose a solution which is good in both respects, i.e., with respect to f^{nom} and with respect to f^{wc}. To find such a solution, we propose the following strategies that can be followed by a decision maker.

A two-stage strategy. In the two-stage strategy, the decision maker may first concentrate on the nominal scenario and identify a most interesting nominal efficient solution $\hat{x}$ based on her/his preferences. This may be done with an interactive method, see e.g., [8, 23, 26]. The interactive solution process also identifies what kind of values of the objective functions are desirable according to the preferences of the decision maker. In the second stage, the decision maker then takes robustness into account as follows: For the identified $\hat{x}$ we compute its price of robustness $\text{price}(\hat{x})$ together with its closest minmax robust efficient solution $\bar{x}$. If a closest solution does not exist we take $\bar{x}$ with $\|\bar{x} - \hat{x}\|_\infty \approx \text{price}(\hat{x})$. Since $\bar{x}$ is the closest solution from X^{wc} to $\hat{x}$ it is likely that it is not too far from the preferences of the decision maker that have been already used in the nominal case. The price of robustness $\text{price}(\hat{x})$ is the nominal quality the decision maker has to sacrifice for changing $\hat{x}$ to this minmax robust efficient solution while still keeping her/his preferences. This value should be compared with the *gain of robustness*

$$\text{gain}(\hat{x}, \bar{x}) = \|f^{\text{wc}}(\hat{x}) - f^{\text{wc}}(\bar{x})\|_\infty$$

when changing from the solution $\hat{x}$ to the (nominal closest) minmax robust efficient solution $\bar{x}$. Being presented the values of $\text{price}(\hat{x})$ and of $\text{gain}(\hat{x}, \bar{x})$, the decision maker can then decide if it is worth to change the nominal solution $\hat{x}$ to $\bar{x}$.

- If $\text{price}(\hat{x})$ is large or if $\text{gain}(\hat{x}, \bar{x})$ is small, the decision maker should keep the nominal efficient solution $\hat{x}$.
- It is preferable to change to $\bar{x}$ if the decision maker is very risk-averse i.e., the decision maker wants to be prepared for the worst case, or if $\text{price}(\hat{x})$ is small, or if $\text{gain}(\hat{x}, \bar{x})$ is large compared to $\text{price}(\hat{x})$.
- If the decision maker does not want to sacrifice that much nominal quality but still wants to increase the robustness of the solution $\hat{x}$, (s)he can specify a maximum tolerable loss ε on the nominal quality by defining $\varepsilon_i \leq \text{price}(\hat{x})$, and solve $(P^{\text{light},\varepsilon})$ to find a lightly robust efficient solution x which
  - is still close to $\hat{x}$, i.e., it keeps the preferences of the decision maker in the nominal scenario,
  - has loss of nominal quality of at most ε,
  - and is the most robust solution among all solutions in $(P^{\text{light},\varepsilon})$, i.e. probably more reliable than $\hat{x}$.

Lexicographic strategies. If the decision maker has no specific preferences but is either mainly interested in the nominal quality or mainly interested in minimizing the risk, it might be appropriate to choose the nominal efficient solution $\hat{x} \in X^{\text{nom}}$ which has the smallest price of robustness (in the first case) or to compute the robust efficient solution $\bar{x} \in X^{\text{wc}}$ which is closest to the set of nominal efficient solutions X^{nom} (in the second case). In mathematical terms we solve

$$\min_{\hat{x} \in X^{\text{nom}}} \min_{\bar{x} \in X^{\text{wc}}} \|f^{\text{nom}}(\hat{x}) - f^{\text{nom}}(\bar{x})\|_\infty$$

and receive a pair of closest points $\hat{x}$ and $\bar{x}$. A risk-averse decision maker (without specific preferences otherwise) might then choose $\bar{x}$ while a decision maker mainly interested in nominal quality, again without specific preferences, can choose $\hat{x}$. The optimization problem can be geometrically solved when the sets X^{wc} and X^{nom} are known, but is otherwise hard to compute.

We illustrate the strategies in an example.

Example 5. *We continue Example 4 to illustrate the two-stage strategy.*

We selected three different nominal efficient solutions (which might reflect the individual preferences for three different decision makers): x^1 *is the lexicographic solution with respect to* f_1^{nom}, x^2 *is some solution in which the decision maker wants to have a good value of* f_2^{nom} *but a not too bad value of* f_1^{nom} *and* x^3 *is the lexicographic solution with respect to* f_2^{nom}. *We computed the price of robustness* $\text{price}(x^l)$, $l = 1, 2, 3$ *as illustrated in Figure 8. The figure shows the closest minmax robust efficient solutions for each of the three selected nominal efficient solutions. In this example, their* price*-values are:* $\text{price}(x^1) = 2$, $\text{price}(x^2) = 1.5$, *and* $\text{price}(x^3) = 0$ *and the corresponding gains are* $\text{gain}(x^1, \bar{x}^1) = 2$, $\text{gain}(x^2, \bar{x}^2) = 1$, *and* $\text{gain}(x^3, \bar{x}^3) = 0$. *Based on the values above, the decision maker can then make the choices described in the strategy. In our case, the decision maker with* x^3 *as her or his most preferred nominal efficient solution might be extremely satisfied because the nominal efficient solution is a minmax robust efficient solution. On the other hand, the decision makers having preferences for* $\hat{x}^1$ *and* $\hat{x}^2$ *observe that* $\text{gain}(\hat{x}^i, \bar{x}^i)$ *is rather low compared to what they would have to pay. If not over-conservative they probably keep the nominal efficient solutions found.*

We finally illustrate the ***lexicographic strategies****. The closest pair of points on the two fronts is depicted in Figure 8 as well. We see that* x^3 *is the nominal efficient solution with the lowest price of robustness (a good choice for a decision maker who mainly cares for nominal quality) while it is also the minmax robust efficient solution which is closest to the set of nominal efficient solutions, i.e., a good choice for a risk-averse decision maker.*

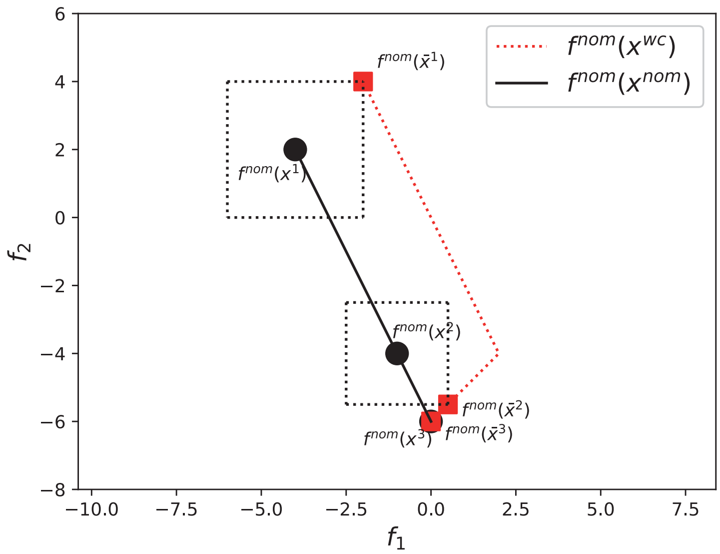


Figure 8: Nominal efficient solutions and their closest minmax robust efficient solutions.

6 Conclusion

In this paper, we formally analyzed nominal efficient solutions, minmax robust efficient solutions, and lightly robust efficient solutions to multiobjective optimization problems with uncertain parameters in the objective functions. We evaluated and compared the three different sets of solutions under the nominal scenario and in the worst case. We found that in the worst case, the set of minmax robust efficient solutions upper dominates the sets of nominal and lightly robust efficient solutions. We also found that in the nominal case, the set of nominal efficient solution upper dominates the set of lightly robust solutions and the set of lightly robust efficient solutions upper and lower dominates the shifted (with respect to ε) outcomes of the lightly robust efficient solutions. We illustrated their relationships with different examples.

In order to further analyze the trade-off between nominal quality and robustness, we proposed a measure for the *price of robustness*. We also analyzed its relationship to lightly robust efficient solutions. For supporting the decision maker to find a solution which is satisfactory in both nominal quality and robustness, we developed two strategies based on the *price of robustness*. We illustrated the utilization of the strategies in an example.

The two strategies rely on the measure *price of robustness* which can be easily computed if the set of minmax robust efficient solutions is known. If the minmax robust efficient solutions cannot be numerically computed, we can approximate the minmax robust efficient front, which can be considered as a future research direction. Our current measure *price of robustness* depends on a fixed nominal efficient solution. Another interesting future research direction is to quantify the *price of robustness* of the whole set of nominal efficient solutions.

Acknowledgments

We thank the mobility grant of the Science Council at the University of Jyvaskyla. We thank Prof. Kaisa Miettinen for useful comments during the preparation of the manuscript.

References

[1] A. Ben-Tal, L. E. Ghaoui, and A. Nemirovski. *Robust Optimization*. Princeton University Press, Princeton and Oxford, 2009.

[2] A. Ben-Tal, A. Goryashko, E. Guslitzer, and A. Nemirovski. Adjustable robust solutions of uncertain linear programs. *Mathematical Programming A*, 99:351–376, 2003.

[3] H. P. Benson and E. Sun. New closedness results for efficient sets in multiple objective mathematical programming. *Journal of Mathematical Analysis and Applications*, 238(1):277–296, 1999.

[4] C. Berge. *Topological Spaces*. Oliver and Boyd, 1963.

[5] D. Bertsimas and M. Sim. The price of robustness. *Operations Research*, 52(1):35–53, 2004.

[6] T. T. Binh and U. Korn. Mobes: A multiobjective evolution strategy for constrained optimization problems. In *proceeding of the third international conference on genetic algorithms*, pages 176–182, 1997.

[7] R. Bokrantz and A. Fredriksson. Necessary and sufficient conditions for pareto efficiency in robust multiobjective optimization. *European Journal of Operational Research*, 262(2):682 – 692, 2017.

[8] J. Branke, K. Ded, K. Miettinen, and R. Slowinski, editors. *Multiobjective Optimization, Interactive and Evolutionary Approaches*. Springer, 2008.

[9] A. Chassein and M. Goerigk. A bicriteria approach to robust optimization. *Computers and Operations Research*, 66:181–189, 2016.

[10] M. Ehrgott, J. Ide, and A. Schöbel. Minmax robustness for multi-objective optimization problems. *European Journal of Operational Research*, 239:17–31, 2014.

[11] M. Fischetti and M. Monaci. Light robustness. In R. K. Ahuja, R. Möhring, and C. Zaroliagis, editors, *Robust and Online Large-scale Optimization*, pages 61–84. Springer, 2009.

[12] J. Fliege and R. Werner. Robust multiobjective optimization & applications in portfolio optimization. *European Journal of Operational Research*, 234(2):422–433, 2014.

[13] M. Goerigk, M. Knoth, M. Müller-Hannemann, M. Schmidt, and A. Schöbel. The Price of Robustness in Timetable Information. In A. Caprara and S. Kontogiannis, editors, *11th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems*, volume 20 of *OpenAccess Series in Informatics (OASIcs)*, pages 76–87, Dagstuhl, Germany, 2011. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

[14] M. Goerigk, M. Knoth, M. Müller-Hannemann, M. Schmidt, and A. Schöbel. The Price of Strict and Light Robustness in Timetable Information. *Transportation Science*, 48:225–242, 2014.

[15] M. Goerigk and A. Schöbel. Algorithm engineering in robust optimization. In L. Kliemann and P. Sanders, editors, *Algorithm Engineering: Selected Results and Surveys*, pages 245–279. 2016.

[16] F. Hassanzadeh, H. Nemati, and M. Sun. Robust optimization for interactive multi-objective programming with imprecise information applied to R&D project portfolio selection. *European Journal of Operational Research*, 238(1):41–53, 2014.

[17] D. Iancu and N. Trichakis. Pareto efficiency in robust optimization. *Management Science*, 60:130–147, 2014.

[18] J. Ide and A. Schöbel. Robustness for uncertain multi-objective optimization: A survey and analysis of different concepts. *OR Spectrum*, 38(1):235–271, 2016.

[19] A. Khan, C. Tammer, and C. Zălinescu. *Set-valued Optimization*. Springer, Berlin Heidelberg, 2015.

[20] K. Kuhn, A. Raith, M. Schmidt, and A. Schöbel. Bicriteria robust optimization. *European Journal of Operational Research*, 252:418–431, 2016.

[21] D. Kuroiwa and G. M. Lee. On robust multiobjective optimization. *Vietnam Journal of Mathematics*, 40(2&3):305–317, 2012.

[22] C. Liebchen, M. Lübbecke, R. H. Möhring, and S. Stiller. The concept of recoverable robustness, linear programming recovery, and railway applications. In R. K. Ahuja, R. Möhring, and C. Zaroliagis, editors, *Robust and online large-scale optimization*, pages 1–27. Springer, 2009.

[23] K. Miettinen. *Nonlinear Multiobjective Optimization*. Kluwer Academic Publishers, 1999.

[24] I. H. Mordechai. The domination property in multicriteria optimization. *Journal of Mathematical Analysis and Applications*, 114(1):7–16, 1986.

[25] A. Schöbel. Generalized light robustness and the trade-off between robustness and nominal quality. *Mathematical Methods of Operations Research*, 80(2):161–191, 2014.

[26] R. E. Steuer. *Multiple Criteria Optimization: Theory, Computation, and Applications*. John Wiley & Sons, Inc., 1986.

[27] M. M. Wiecek and G. M. Dranichak. Robust multiobjective optimization for decision making under uncertainty and conflict. In A. Gupta, A. Capponi, J. C. Smith, and H. J. Greenberg, editors, *Optimization Challenges in Complex, Networked and Risky Systems*, pages 84–114, 2016.

PV

DECISION MAKING IN MULTIOBJECTIVE OPTIMIZATION PROBLEMS UNDER UNCERTAINTY: BALANCING BETWEEN ROBUSTNESS AND QUALITY

by

Yue Zhou-Kangas and Kaisa Miettinen

Submitted to a journal

Decision making in multiobjective optimization problems under uncertainty: balancing between robustness and quality

Yue Zhou-Kangas   Kaisa Miettinen
University of Jyvaskyla, Faculty of Information Technology
P.O. Box 35 (Agora), FI-40014 University of Jyvaskyla, Finland
{yue.y.zhou-kangas, kaisa.miettinen}@jyu.fi

September 7, 2018

Abstract

As an emerging research field, multiobjective robust optimization employs minmax robustness as the most commonly used concept. Light robustness is a concept in which a parameter, tolerable degradations, can be used to control the loss in the objective function values in the most typical scenario for gaining in robustness. In this paper, we develop a lightly robust interactive multiobjective optimization method, LiRoMo, to support a decision maker to find a most preferred lightly robust efficient solution with a good balance between robustness and the objective function values in the most typical scenario. In LiRoMo, we formulate a lightly robust subproblem utilizing an achievement scalarizing function which involves a reference point specified by the decision maker. With this subproblem, we compute lightly robust efficient solutions with respect to the decision maker's preferences. With LiRoMo, we support the decision maker in understanding the lightly robust efficient solutions with an augmented value path visualization. We use two measures 'price to be paid for robustness' and 'gain in robustness' to support the decision maker in considering the trade-offs between robustness and quality. As an example to illustrate the advantages of the method, we formulate and solve a simple investment portfolio optimization problem.

Keywords: Multiobjective robust optimization, Interactive methods, Light robust efficiency, Handling uncertainty, Trade-off between robustness and quality, Decision support.

1 Introduction

Many decision-making problems involve multiple criteria to be optimized and they also include uncertainty from different sources such as uncertain future developments and imprecise measurements. Due to the uncertainty, the outcome of implementing a decision can become unexpected and undesired. In recent years, both researchers and practitioners have started to pay attention to dealing with multiple criteria and the involved uncertainty simultaneously. The approaches employed depend on the availability of data and the expert knowledge of the decision maker. When enough data about the uncertainty is available, problems can be solved with stochastic approaches (see e.g., [20]) and when the expert judgments on fuzzy memberships can be relied on, fuzzy approaches can be implemented (see e.g., [23]). On the other hand, when there is no sufficient data available or the decision maker does not have sufficient knowledge to judge the memberships, robust optimization approaches can be utilized (see e.g., [22, 36]).

In robust optimization approaches, typically the uncertainty is modeled as parameters whose exact values are not known but are assumed to stem from a set. We call this set an uncertainty set. Possible realizations of the unknown parameters are called scenarios, which are the elements in the uncertainty set. We call the most typical or expected scenario the nominal scenario and the objective function values in the nominal scenario as the nominal quality. In order to find solutions when uncertainty is taken into account, different robustness concepts have been developed. Flimsly robust solutions [8, 22] are the best solutions in one of the possible realizations of the uncertain parameters. Highly robust solutions [8, 12, 17, 22] are the best solutions in all the possible realizations of uncertain parameters at the same time. The most common concepts used in multiobjective robust optimization belong to the family of minmax robustness concepts (e.g., [9, 13, 14, 25]). For minmax robustness, we optimize the objective functions in the worst case over all scenarios. The solutions computed are said to be minmax robust efficient.

However, minmax robust efficient solutions are not always easy to compute. In addition, the conservatism of minmax robustness has been recognized in single objective cases (see e.g., [5, 7, 30]), i.e., the objective function value of a minmax robust solution is usually not that good in other scenarios. Also, the decision maker may not be willing to make decisions based on the worst possible realizations of the uncertain parameters. On the other hand, if a solution is found only by optimizing in the nominal scenario, the possibility of realizations of other scenarios is ignored. So, the decision maker may want to focus on the nominal scenario but make decisions which are still valid in other scenarios. Based on the consideration of both robustness and the nominal quality, developments in single objective cases have led to different concepts in controlling the degradation of the objective function value in the nominal scenario (see e.g., [3, 4, 5, 7, 18, 30]). These concepts share the same idea of balancing between robustness and the nominal quality.

Similar thoughts have initiated developments in multiobjective optimization in [21, 22, 24]. In [21], a multiobjective version of the concept proposed in [7] has been developed and the robust optimization problem is solved based on the augmented Chebychev function (see e.g., [34]). The concept of light robustness has been originally proposed in [15] for single objective linear problems with uncertain parameters which stem from an interval uncertainty set and generalized in [30] for other types of uncertain sets and quasi-convex objective functions. It has been extended to light robust efficiency for multiobjective cases in [22]. The idea of light robust efficiency is to find a robust solution with tolerable degradations in the nominal quality. Concepts based on light robustness have been developed in [24] for problems where uncertain parameters are involved in one of the two objectives in bi-objective optimization problems. An algorithm for bi-objective combinatorial optimization problems with a finite uncertainty set has also been developed. Solution methods for supporting a decision maker to find a most preferred lightly robust efficient solution when uncertainty is involved in multiple objectives have not been published.

In light robustness, the loss in the nominal quality is controlled by a parameter, indicating tolerable degradations in nominal quality. With respect to the tolerable degradations in the nominal quality, we optimize in the worst case to seek for a most robust solution. By doing so, usually, the computed solutions do not have as good quality in the worst case as minmax robust efficient solutions but they are usually better in terms of nominal quality. It is proven in [22] that with a sufficiently large tolerance on the degradations in the nominal quality, a computed lightly robust efficient solution is a minmax robust efficient solution. The relationships between minmax robust efficient, lightly robust efficient and nominal efficient solutions are analyzed in [38]. It is also proven that the lightly robust efficient solutions are good trade-offs between robustness and the nominal quality. Thus, with the concept of light robust efficiency, the decision maker can focus on the nominal quality but find solutions that are also valid in other scenarios. This is why we focus on utilizing the concept of light robustness.

In the solution process of a multiobjective optimization problem, the decision maker can utilize the parameter, tolerable degradations in the nominal quality, to control the trade-off between robustness and the nominal quality in order to find a solution with a good balance in both respects. As there usually is a set of efficient solutions, the decision maker needs support to understand the trade-offs among the objectives to choose a final solution. When robustness is considered in the solution process, the decision maker needs further support, not only for the trade-offs among the objectives but also the trade-offs between robustness and the nominal quality.

In this paper, we focus on applying light robustness for decision support and develop an interactive method, LiRoMo, to support the decision maker to find a good balance between robustness and the nominal quality by finding a most preferred lightly robust efficient solution. Interactive methods (see e.g., [26, 33]) are a category of solution methods for multiobjective optimization problems. With interactive methods, the interactive solution process starts by presenting an initial solution to the decision maker. If the decision maker is not satisfied, (s)he can specify preferences for a more desired solution. Then, a scalarized subproblem is solved to find a new solution which satisfies the preferences as well as possible. This process continues until the decision maker finds a most preferred solution. During the interactive solution process, the decision maker can learn about the problem, for example, some specific properties of the problem. The decision maker can also learn about the attainable objective function values and at the same time learn how achievable her or his own preferences are.

When seeking a most preferred lightly robust efficient solution, the decision maker needs to consider the robustness and the nominal quality of solutions at the same time. So, (s)he should be provided the opportunity not only to learn about the problem, the attainable objective function values, and her or his own preferences but also to learn about the trade-offs between robustness and the nominal quality. With an interactive method, we can provide such support. As a result, the decision maker can eventually find a final solution with a satisfactory balance between robustness and the nominal quality.

The decision maker directs the interactive solution process and we need to generate solutions utilizing the decision maker's preferences. As a common approach to compute efficient solutions in multiobjective optimization, scalarization functions combine multiple objectives (and preferences) into a single objective such that an optimal solution of the scalarized problem is an efficient solution to the multiobjective optimization problem. Scalarization for computing minmax robust efficient solutions has been discussed in [9]. In the LiRoMo method, we formulate a lightly robust subproblem based on an achievement scalarization function [37]. For computing lightly robust efficient solutions efficiently, we reformulate the subproblem by utilizing the properties of the objective functions and the uncertainty sets. For supporting the decision maker to understand the trade-offs between robustness and the nominal quality, we quantify the gain in robustness and the price to be paid for robustness. In order to effectively illustrate the solutions to the decision maker, we augment the value path visualization [27] to support the decision maker.

The rest of the paper is organized as follows: Section 2 describes the definitions and background information needed which is followed by Section 3 where we formulate the lightly robust subproblem based on the achievement scalarizing function and present its reformulation. In Section 4, we describe the LiRoMo method for supporting the decision-making process of balancing between robustness and quality. Then we formulate and solve an investment portfolio optimization problem to demonstrate the application of the LiRoMo method in Section 5 and conclude in Section 6.

2 Preliminaries

2.1 Multiobjective optimization under uncertainty

We consider multiobjective optimization problems where some parameters in the objective functions are unknown but stem from an uncertainty set $\mathcal{U}$ in the following form:

$$\begin{pmatrix} \text{minimize} & (f_1(x,\xi), \cdots, f_k(x,\xi)) \\ \text{subject to} & x \in X \end{pmatrix}_{\xi \in \mathcal{U}}, \tag{1}$$

where $x = (x_1, \cdots, x_n)^T$ is the decision vector which belongs to a feasible set $X \in \mathbb{R}^n$ and ξ represents the uncertain parameters which stem from the set $\mathcal{U}$. We assume that a nominal scenario $\hat{\xi}$ is known. When robustness does not play a role, the problem is solved in the nominal scenario as a deterministic problem:

$$\begin{aligned} \text{minimize} \quad & \left(f_1(x,\hat{\xi}), \cdots, f_k(x,\hat{\xi})\right) \\ \text{subject to} \quad & x \in X. \end{aligned} \tag{2}$$

We call problem (2) a nominal problem and for each $x \in X$, we define the objective vector $f^{\text{nom}}(x) := (f_1(x,\hat{\xi}), \cdots, f_k(x,\hat{\xi}))^T$ as its nominal quality. We say that x^* is an efficient (or Pareto optimal) solution to (2), if there does not exist any $x' \in X$ such that $f_i^{\text{nom}}(x') \leq f_i^{\text{nom}}(x^*)$ for all $i = 1, ..., k$ and $f_j^{\text{nom}}(x') < f_j^{\text{nom}}(x^*)$ for at least one j. The set of efficient solutions to (2) is denoted by X^{nom}. Problem (2) can be solved with methods for deterministic multiobjective optimization problems in the following form where no uncertainty is involved:

$$\begin{aligned} \text{minimize} \quad & (f_1(x), \cdots, f_k(x)) \\ \text{subject to} \quad & x \in X. \end{aligned} \tag{3}$$

For decision making, it is usually useful for the decision maker to know the ranges of the objective function values in X^{nom}. The ideal objective vector z^{ideal} and the nadir objective vector z^{nad} can provide information on the ranges. The ideal objective vector is formed by the individual optima of each objective function. The nadir objective vector can give the upper bounds of the objective vectors correspond to X^{nom}. The nadir objective vector can be approximated by for example the payoff table (see e.g., [26, 33]). This approximated vector can over or underestimate the nadir objective vector. There are also other ways to approximate the nadir objective vector (see e.g.,[11]). For computational reasons, we also define a utopian objective vector $z^{\text{uto}} = (z_1^{\text{ideal}} - a, \cdots, z_k^{\text{ideal}} - a)^T$, where $a > 0$ is a small scalar.

2.2 Minmax robustness

Minmax robust efficient solutions are found by optimizing in the worst case. The values of the uncertain parameters with which a solution $x \in X$ attains its worst objective function values are called the worst case scenarios or simply worst cases. For a fixed $x \in X$, finding its worst case objective vector(s) requires maximizing k objectives simultaneously. If the problem is objective-wise uncertain, i.e., the uncertain parameters in the objective functions do not relate to each other (see [13] for a formal definition), there exists a single worst case scenario. Otherwise, there usually exists a set of worst case scenarios.

In this paper we employ the concept called point-based minmax robustness (see [16, 25]) for optimizing in the worst case. A point-based minmax robust optimization problem is defined as follows:

$$\begin{aligned} \text{minimize} \quad & \left(\sup_{\xi \in \mathcal{U}} f_1(x,\xi), \cdots, \sup_{\xi \in \mathcal{U}} f_k(x,\xi)\right) \\ \text{subject to} \quad & x \in X. \end{aligned} \tag{4}$$

In this formulation, with a fixed $x \in X$, the worst case objective function values are represented by an objective vector. This vector is formed by the individual maxima of each objective function with respect to the uncertain parameters involved. So, we consider a single objective vector in the solution process of problems regardless of if the problem is objective-wise uncertain or not. However, the point-based worst case usually does not realize i.e., the objective functions do not attain their worst values simultaneously unless the problem is objective-wise uncertain. We use $f^{\text{wc}}(x) := (f_1(x, \bar{\xi}), \cdots, f_k(x, \bar{\xi}))^T$ to represent a point-based minmax robust objective vector for a fixed $x \in X$. The vector f^{wc} consists of the individual maxima of each objective function. We refer to f^{wc} as the worst case.

If the decision maker wishes to concentrate only on robustness, we can solve problem (4) to find point-based minmax robust efficient solutions for her or him. For helping the decision maker to understand the ranges of the objective function values of the point-based robust efficient solutions, we can identify the robust ideal objective vector $z^{\text{ideal,wc}}$ and the robust nadir objective vector $z^{\text{nad,wc}}$. The robust ideal objective vector $z^{\text{ideal,wc}}$ is formed by the individual minima of the objective functions in problem (4) and $z^{\text{nad,wc}}$ can be approximated by the payoff table. Corresponding to the nominal utopian objective vector, we also have the robust utopian objective vectors $z^{\text{uto,wc}} = (z_1^{\text{ideal,wc}} - a, \cdots, z_k^{\text{ideal,wc}} - a)^T$, where $a > 0$ is a small scalar.

2.3 Light robustness

In the concept of light robustness, we assume that the set X^{nom} is nonempty. The idea is that we first determine some $\hat{x} \in X^{\text{nom}}$, and then we look for the most robust solution (i.e., optimize in the worst case) with tolerable degradations in the nominal quality. The tolerable degradations are given by $\varepsilon \in \mathbb{R}^k$, whose component ε_i represents the allowed degradation of f_i^{nom}.

For each $\hat{x} \in X^{\text{nom}}$, the point-based lightly robust problem with respect to (1) can be defined as follows:

$$
\begin{array}{ll}
\text{minimize} & \left(\sup\limits_{\xi \in \mathcal{U}} f_1(x, \xi), \cdots, \sup\limits_{\xi \in \mathcal{U}} f_k(x, \xi) \right) \\
\text{subject to} & x \in X \\
& f_i^{\text{nom}}(x) \leq f_i^{\text{nom}}(\hat{x}) + \varepsilon_i \text{ for all } i = 1, \cdots, k.
\end{array} \tag{5}
$$

In this formulation, we refer to $F^{\text{light}}(\hat{x}, \varepsilon) := \{x \in X : f_i^{\text{nom}}(x) \leq f_i^{\text{nom}}(\hat{x}) + \varepsilon_i \text{ for all } i = 1, \cdots, k\}$ as the lightly robust feasible set. Lightly robust efficient solutions are identified by optimizing in the worst case with respect to the lightly robust feasible set. We say that a solution $x^{\text{light},*} \in F^{\text{light}}(\hat{x}, \varepsilon)$ is lightly robust efficient if there does not exists any solution $x^{\text{light},**} \in F^{\text{light}}(\hat{x}, \varepsilon)$ such that $f_i^{\text{wc}}(x^{\text{light},**}) \leq f_i^{\text{wc}}(x^{\text{light},*})$ for all $i = 1, \cdots, k$ and $f_j^{\text{wc}}(x^{\text{light},**}) < f_j^{\text{wc}}(x^{\text{light},*})$ for at least one j. The set of lightly robust efficient solutions is denoted by $X^{\text{light},\varepsilon}$.

By varying the value of ε, we can alter the trade-offs between robustness and the nominal quality in (5). Figure 1 illustrates the idea of light robustness with two objectives. The dashed line represents the set $f^{\text{nom}}(X^{\text{nom}})$. The dot represents the nominal objective vector $f^{\text{nom}}(\hat{x})$ of the pre-selected nominal efficient solution $\hat{x}$. With a given ε, the area which can contain the nominal objective vectors of lightly robust efficient solutions is indicated by the dotted lines.

In [38], the gain in robustness and the price to be paid for robustness are defined for fixed nominal efficient and lightly robust efficient solutions. With fixed $\hat{x} \in X^{\text{nom}}$ and $x^{\text{light},\varepsilon} \in X^{\text{light},\varepsilon}$, we can quantify the gain in robustness by comparing their corresponding objective vectors in the worst case with the following measure:

$$
gain(x^{\text{light},\varepsilon}, \hat{x}) = \| f^{\text{wc}}(\hat{x}) - f^{\text{wc}}(x^{\text{light},\varepsilon}) \|_\infty. \tag{6}
$$

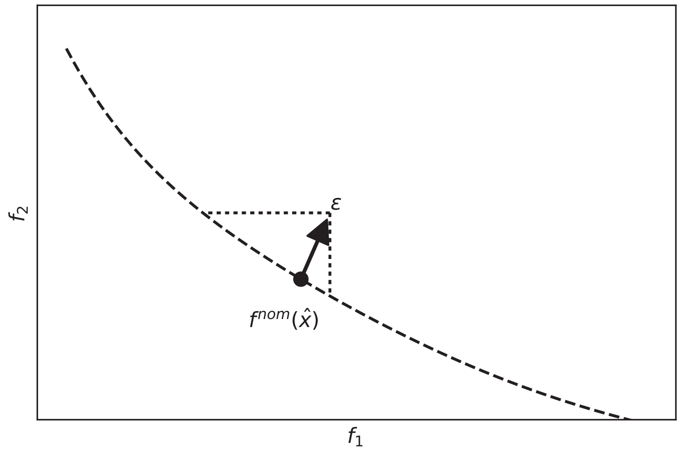


Figure 1: Light robustness

The notation $\|\cdot\|_\infty$ represents the L_∞ norm, which tells the maximum gain in robustness among the objectives. Other norms can also be used, for example, $\|\cdot\|_1$ can give the value on the average gain in robustness among the objectives. Analogously, with fixed $\hat{x} \in X^{\text{nom}}$ and $x^{\text{light},\varepsilon} \in X^{\text{light},\varepsilon}$, we can quantify the price to be paid for robustness by comparing their corresponding objective vectors in the nominal scenario with the following measure:

$$price(x^{\text{light},\varepsilon}, \hat{x}) = \|f^{\text{nom}}(\hat{x}) - f^{\text{nom}}(x^{\text{light},\varepsilon})\|_\infty. \quad (7)$$

2.4 Achievement scalarizing function

As mentioned earlier, one approach to solving multiobjective optimization problems in the form of (3) is to formulate a single objective optimization subproblem by scalarizing. The achievement scalarizing function [37] is one of the widely used scalarizing functions. In this paper, we utilize the following subproblem based on a variant of the achievement scalarizing function:

$$\begin{aligned} &\text{minimize} && \alpha + \rho \sum_{i=1}^{k} w_i(f_i(x) - \bar{z}_i) \\ &\text{subject to} && w_i(f_i(x) - \bar{z}_i) \leq \alpha \text{ for all } i = 1, \cdots, k \\ &&& x \in X, \end{aligned} \quad (8)$$

where ρ is a small scalar bounding the trade-offs among the objectives, $\bar{z}$ is a reference point and its components $\bar{z}_i$ are aspiration levels which represent the desired values of the objective function f_i given by the decision maker. We have presented subproblem (8) in a differentiable form (assuming that the objective functions are differentiable), where the auxiliary variable α is used for the transformation from a minmax form (see e.g., [26]). The positive weight vector w sets a direction which the reference point is projected onto the set of objective vectors corresponding to the efficient solutions. We refer to w as projection direction.

As discussed in the literature (e.g., in [10, 26, 37]), any optimal solution of (8) is an efficient solution for (2) and any efficient solution with trade-offs bounded by ρ can be found by changing $\bar{z}$. Such solutions are also called properly efficient solutions (for details, see e.g., [37]). The achievement scalarizing function works independently of the problem and preferences considered. For example, the reference point can be feasible or infeasible and the problem can be convex or nonconvex.

3 Computing lightly robust efficient solutions

3.1 Lightly robust subproblem based on the achievement scalarizing function

In this section, we assume that $\max_{\xi\in\mathcal{U}} f(x,\xi)$ exists for all $i = 1, \cdots, k$ for a fixed $x \in X$. The existence of $\max_{\xi\in\mathcal{U}} f(x,\xi)$ for all $i = 1, \cdots, k$ can be guaranteed for example when X is finite, $\mathcal{U}$ is compact and $f(x,\cdot) : \mathcal{U} \to \mathbb{R}^k$ is continuous in ξ for every fixed $x \in X$. Based on (5) and (8), a lightly robust subproblem based on achievement scalarizing function can be given as follows:

$$
\begin{array}{ll}
\text{minimize} & \alpha + \rho \sum_{i=1}^{k} w_i(\max_{\xi\in\mathcal{U}} f_i(x,\xi) - z_i) \\
\text{subject to} & f_i(x,\hat{\xi}) \le f_i(\hat{x},\hat{\xi}) + \varepsilon_i \text{ for all } i = 1, \cdots, k \\
 & w_i(\max_{\xi\in\mathcal{U}} f_i(x,\xi) - z_i) \le \alpha \text{ for all } i = 1, \cdots, k \\
 & x \in X.
\end{array} \tag{9}
$$

Theorem 1. *Any optimal solution of* (9) *is a lightly robust efficient solution for* (1).

Proof. Let x^* be an optimal solution of (9) and assume that it is not lightly robust efficient to problem (1). Then there exists $x' \in F^{\text{light}}(\hat{x},,\varepsilon)$, such that $f_i^{\text{wc}}(x') \le f_i^{\text{wc}}(x^*)$ for all $i = 1, ..., k$ and $f_j^{\text{wc}}(x') < f_j^{\text{wc}}(x^*)$ for at least one j. So we have $\max_i[w_i(\max_{\xi\in\mathcal{U}} f_i(x',\xi) - z_i)] + \rho \sum_{i=1}^{k}(\max_{\xi\in\mathcal{U}} f(x',\xi) - z_i) < \max_i[w_i(\max_{\xi\in\mathcal{U}} f_i(x^*,\xi) - z_i)] + \rho \sum_{i=1}^{k}(\max_{\xi\in\mathcal{U}} f(x^*,\xi) - z_i)$. This contradicts with the assumption that x^* is an optimal solution to (9). Thus x^* is a lightly robust efficient solution to (1). □

The formulation (9) involves finding the optimal solution of the maximization problem to identify the worst case value of each objective function, i.e., the problem $\max_{\xi\in\mathcal{U}} f_i(x,\xi)$ for a fixed $x \in X$. So, using (9) to find a lightly robust solution requires solving a more complicated optimization problem than for a deterministic problem with (8). As mentioned in the literature (e.g., in [4, 6]), problem (9) cannot be efficiently solved in a general case. Next, we reformulate the problem so that it can be efficiently solved by making some assumptions.

3.2 Reformulating the lightly robust subproblem

Reformulating problems to compute robust optimal solutions is a research topic with a long history in single objective optimization (see e.g., [2, 7, 32]).

Here we utilize the properties of $f_i(x,\xi)$ and $\mathcal{U}$. The following result have been presented in Corollary 2.14 in [35]:

Lemma 2. *A real-valued function $g(x)$ on a compact convex set C attains its maximum at an extreme point of C.*

We consider the uncertainty set $conv(\mathcal{U})$ which is called a polyhedral uncertainty set. It is defined by a set of scenarios $\mathcal{U} = \{\xi^1, \cdots, \xi^m\}$ as the extreme points of the convex hull. The results in Lemma 2 has been utilized in [24] for reducing polyhedral uncertainty sets to discrete uncertainty sets involved in one of the objective functions in bi-objective optimization problems. It has also been utilized in [13] for replacing $conv(\mathcal{U})$ by $\mathcal{U}$ for objective-wise uncertain problems. Since in this paper, we optimize in the point-based worst case for finding lightly robust efficient solutions, problem (9) has the following equivalent reformulation:

Theorem 3. *Let $conv(\mathcal{U})$ be a polyhedral uncertainty set with extreme points $\mathcal{U} = \{\xi^1, ..., \xi^m\}$ and f_i quasi-convex in ξ for every fixed $x \in X$ and $i = 1, \cdots, k$. Problem (9) can be solved in the following form:*

$$\begin{array}{ll} \textit{minimize} & \alpha + \rho \sum_{i=1}^{k} w_i(\gamma - \bar{z}_i) \\ \textit{subject to} & x \in X \\ & f_i(x, \hat{\xi}) \leq f_i(\hat{x}, \hat{\xi}) + \varepsilon \textit{ for all } i = 1, \cdots, k \\ & w_i(\gamma - \bar{z}_i) \leq \alpha \\ & f_i(x, \xi^j) \leq \gamma \textit{ for all } i = 1, \cdots, k \textit{ and } j = 1, \cdots, m. \end{array} \tag{10}$$

Proof. Based on Lemma 2, f_i attains its maximum over $conv(\mathcal{U})$ at an extreme point of it with the assumption that f_i is quasi-convex in ξ for every fixed x. So we have

$$\max_{\xi \in conv(\mathcal{U})} f_i(x, \xi) = \max_j f_i(x, \xi^j).$$

Corresponding to using α in (8), we use the auxiliary variable γ (which is a scalar valued variable) to bound $\max_i \max_j f_i(x, \xi^j)$, i.e., we can solve (9) by considering the extreme points of $conv(\mathcal{U})$ and the objective function with gives the largest value. □

Based on e.g.,[24], polyhedral uncertainty sets include interval uncertainty sets (see e.g., [15, 30]) as special cases. When the uncertain parameters vary in intervals, we obtain a box as the uncertainty set, which is a special polyhedron. On the other hand, affine objective functions are also quasi-convex. So, (10) is also valid for problems with such characteristics.

If we have a multiobjective optimization problem with ξ stemming from a set $\mathcal{U}$ with a finite number of scenarios, i.e., $\mathcal{U}$ is a finite uncertainty set, we can use the reformulation (10) to solve the problem. In this case, we do not need to assume that f_i is quasi-convex in ξ for every fixed $x \in X$ since we can directly compare the objective vectors in the scenarios.

4 The LiRoMo method

As introduced earlier, finding minmax robust efficient or nominal efficient solutions may not well serve the purpose of considering both robustness and the nominal quality. Minmax robust efficient solutions can have bad objective function values in other scenarios and the decision maker may not want to make decisions based on the worst possible realizations of the uncertain parameters. Nominal efficient solutions only concentrate on the nominal quality and other scenarios are ignored. Lightly robust efficient solutions can have both aspects incorporated. In this section, we propose the LiRoMo method to support the decision maker to find a most preferred lightly robust efficient solution with a preferred balance between robustness and the nominal quality. We first discuss the LiRoMo method in general. Then, we present its steps followed by a detailed discussion on the steps.

Using a reference point to find a most preferred efficient solution for the decision maker in an interactive method has been used in earlier research for deterministic problems for example in [28, 37]. In the LiRoMo method, we also utilize reference points as a means for the decision maker to express preference information. We first find a nominal efficient solution $\hat{x}$ which satisfies the preferences of the decision maker as well as possible. This is done by solving (8) based on the reference point specified by the decision maker. Then, based on the tolerable degradations of the nominal quality, which are also specified by the decision maker, we optimize in the worst case to find a lightly robust efficient solution by solving (9).

When the decision maker studies a lightly robust efficient solution, (s)he needs to understand its essence with respect to both robustness and the nominal quality. For the nominal quality, the decision maker needs to consider the nominal objective function values of the lightly robust efficient solution. For robustness, the concept of light robustness only finds the most robust solution with respect to the tolerable degradations. It is not enough to purely rely on this fact. Without additional information, it is hard for the decision maker to understand the trade-off between robustness and nominal quality, i.e., if the robustness gained is worthy of the sacrifice in the nominal quality.

Thus, we provide some additional information to help the decision maker to understand the trade-off between robustness and the nominal quality. In order to provide this information, we can utilize the gain in robustness to compare the lightly robust efficient and nominal efficient solutions in the worst case and use the price to be paid for robustness to compare the two solutions in the nominal scenario. As the objective function values can have very different scales, the original forms of gain in robustness and price to be paid for robustness presented in (6) and (7) need to be normalized. In the LiRoMo method, we calculate the gain in robustness as

$$gain(x^{\text{light},\varepsilon}, \hat{x}) = \left\| \left(\frac{f_1^{\text{wc}}(x^{\text{light},\varepsilon}) - f_1^{\text{wc}}(\hat{x})}{z_1^{\text{nad,wc}} - z_1^{\text{uto,wc}}}, \cdots, \frac{f_k^{\text{wc}}(x^{\text{light},\varepsilon}) - f_k^{\text{wc}}(\hat{x})}{z_k^{\text{nad,wc}} - z_k^{\text{uto,wc}}} \right)^T \right\|_\infty . \tag{11}$$

The value of $gain(x^{\text{light},\varepsilon}, \hat{x})$ quantifies how much better the lightly robust efficient solution is in the worst case compared to the nominal efficient solution. The value of $gain(x^{\text{light},\varepsilon}, \hat{x})$ represents the largest percentage that $x^{\text{light},\varepsilon}$ is better in terms of the worst case objective function values than $\hat{x}$ in the ranges of the objective vectors corresponding to the minmax robust efficient solutions. In addition, we also calculate the price to be paid for robustness:

$$price(x^{\text{light},\varepsilon}, \hat{x}) = \left\| \left(\frac{f_1^{\text{nom}}(x^{\text{light},\varepsilon}) - f_1^{\text{nom}}(\hat{x})}{z_1^{\text{nad}} - z_1^{\text{uto}}}, \cdots, \frac{f_k^{\text{nom}}(x^{\text{light},\varepsilon}) - f_k^{\text{nom}}(\hat{x})}{z_k^{\text{nad}} - z_k^{\text{uto}}} \right)^T \right\|_\infty . \tag{12}$$

The value of $price(x^{\text{light},\varepsilon}, \hat{x})$ quantifies how much worse the lightly robust efficient solution is in the nominal scenario compared to the nominal efficient solution. The value of $price(x^{\text{light},\varepsilon}, \hat{x})$ presents the largest percentage that $x^{\text{light},\varepsilon}$ is worse than $\hat{x}$ in terms of the nominal objective function values in the ranges of the objective vectors corresponding to the nominal efficient solutions. If $price(x^{\text{light},\varepsilon}, \hat{x})$ is larger than $gain(x^{\text{light},\varepsilon}, \hat{x})$, more nominal quality is sacrificed compared to the gain in robustness. By combining the comparison of the two measures and the nominal objective vectors of the lightly robust efficient solution, the decision maker can consider her or his preferences in both respects and make informed decisions.

After understanding the presented lightly robust efficient solution, the decision maker needs to consider two different types of preference information for finding a more desired lightly robust efficient solution. First, (s)he needs to consider the preferences for a more interesting nominal efficient solution. Second, (s)he needs to consider the tolerable degradations in the nominal quality. The decision maker could also consider how much (s)he wants to gain in robustness by the maximum tolerable loss in the nominal quality which is specified by the tolerable degradations.

From the computation point of view, incorporating preferences on the gain in robustness can be achieved by adding additional constraints to the lightly robust problem. However, from a decision-making point of view, the decision maker should know the attainable objective function values and the trade-off between robustness and nominal quality very well. For example, with an unrealistic preference in the tolerance and gain, it may happen that there are no feasible lightly robust efficient solutions because the decision maker sacrifices too little but hopes to gain too much. This is why we do not request the preference information on the gain in robustness from the decision maker in LiRoMo but concentrate on finding a satisfactory lightly robust efficient solution by allowing the decision maker to alter her or his preferences in the nominal objective vector in the form of reference points and specifying the tolerable degradations.

The steps of the LiRoMo method are the followings:

Initialization. Present z^{ideal}, z^{nad} and $z^{\text{ideal,wc}}$ and $z^{\text{nad,wc}}$ to the decision maker. Set the iteration counter $c = 0$.

Step 1. Ask the decision maker to specify the desired values of each objective function which forms the reference point $\bar{z}$.

Step 2. Solve (8) to find a nominal efficient solution $\hat{x}^c$.

Step 3. Present the objective vector corresponding to $\hat{x}^c$ to the decision maker.

Step 4. Ask the decision maker to specify her or his preferences on how much (s)he is willing to sacrifice in the nominal quality to gain robustness which forms the tolerable degradations ε for the preferred nominal solution.

Step 5. With $\hat{x}^c$ and ε, solve (9) to find a lightly robust efficient solution $x^{(\text{light},\varepsilon)_c}$ and compute the gain in robustness $gain(x^{(\text{light},\varepsilon)_c}, \hat{x}^c)$ and the price to be paid for robustness $price(x^{(\text{light},\varepsilon)_c}, \hat{x}^c)$.

Step 6. Present the nominal objective vectors corresponding to $x^{(\text{light},\varepsilon)_c}$ and $\hat{x}^c$ together with the values of $gain(x^{(\text{light},\varepsilon)_c}, \hat{x}^c)$ and $price(x^{(\text{light},\varepsilon)_c}, \hat{x}^c)$ to the decision maker.

Step 7. If the decision maker is satisfied, terminate the solution process and set $x^{(\text{light},\varepsilon)_c}$ as the final solution. Otherwise, continue.

Step 8. If the decision maker wishes to alter the trade-offs between robustness and the nominal quality, i.e., modify the tolerable degradations, ask the decision maker to give a new preferred value of ε, set $c = c + 1$ and go to step 3. If the decision maker wants to change her or his preferences on the nominal efficient solution, set $c = c + 1$ and go to step 1.

In the initialization step, the nominal ideal and nadir objective vectors are presented to the decision maker to help her or him to have a general idea of the ranges of the objective function values among the nominal solutions. This helps the decision maker to specify the reference point in step 1. The robust ideal objective vector and the robust nadir objective vector are also presented to help the decision maker to grasp a picture on the ranges of the objective function values of solutions in the worst case if (s)he fully concentrates on robustness. This can help her or him to specify the tolerable degradations in the nominal quality later. In step 2, we solve (8) to find the nominal efficient solution $\hat{x}^c$ using $\bar{z}$ as the reference point. Then, we can present the objective vector corresponding to $\hat{x^c}$ to the decision maker in step 3 and ask the decision maker to specify her or his preferences on ε in step 4.

In step 5, with $\hat{x}^c$ and ε, we solve (9) to find a lightly robust efficient solution $x^{(\text{light},\varepsilon)_c}$. The value of ε is closely related to the reservation levels (see e.g., [19]). Reservation levels are limits of objective function values that the decision maker cannot accept to go

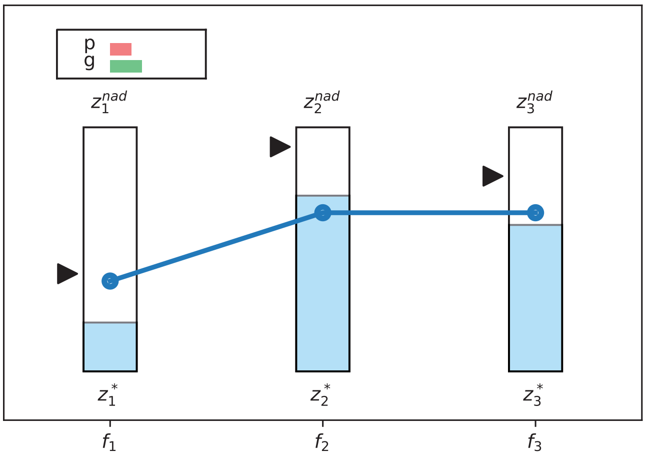


Figure 2: Visualizing a lightly robust efficient solutions

beyond. In our context, the decision maker wants to avoid any nominal objective vector worse than $f^{\text{nom}}(\hat{x})+\varepsilon$. So, with a fixed $f^{\text{nom}}(\hat{x})$, if a decision maker changes the value of ε, (s)he changes the limits of the acceptable nominal objective vector beyond which the corresponding solutions should be avoided. This means that changing the value of ε can be understood as changing the reservation levels.

When solving (9), we use $f^{\text{nom}}(\hat{x}^c)+\varepsilon$ as the reference point. This is because of two reasons. First, the value of $f^{\text{nom}}(\hat{x}^c)+\varepsilon$ is still acceptable for the decision maker. Second, (s)he is expecting the most robust solution within the range of $f^{\text{nom}}(\hat{x}^c)$ and $f^{\text{nom}}(\hat{x}^c)+\varepsilon$, so (s)he is willing to sacrifice until the limits of the acceptable nominal objective function values to gain robustness. We set the projection vector $w = \frac{1}{f^{\text{nom}}(\hat{x}^c)+\varepsilon-f^{\text{nom}}(\hat{x}^c)} = \frac{1}{\varepsilon}$ because in the ideal situation, $\hat{x}^c$ is a lightly robust efficient solution. If $\hat{x}^c$ is not a lightly robust efficient solution, we may preserve the preferences on the trade-offs among the objectives with this projection direction.

In order to efficiently solve (9), we can use the reformulation presented in the subproblem (10). Correspondingly, we can compare the objective function values in the scenarios which specify $\mathcal{U}$ to find the point-based worst case objective vector for computing the value of $gain(x^{(\text{light},\varepsilon)_c}, \hat{x}^c)$. For computing the value of $price(x^{(\text{light},\varepsilon)_c}, \hat{x}^c)$, we only need to evaluate the two solutions in the nominal scenario. If the problem does not meet the assumptions in the reformulation, it is possible to represent the uncertainty set by a set of discrete scenarios by utilizing a good sampling technique presented e.g., in [31]. When we replace the uncertainty set by a set of sampled discrete scenarios, we can use reformulation presented in (10).

In step 6, we visualize $f^{\text{nom}}(x^{(\text{light},\varepsilon)_c})$ and $f^{\text{nom}}(\hat{x}^c)$ together with the values of $gain(x^{(\text{light},\varepsilon)_c}, \hat{x}^c)$ and $price(x^{(\text{light},\varepsilon)_c}, \hat{x}^c)$ to the decision maker. As mentioned earlier, by combining the two types of information, the decision maker can understand the trade-offs in the objectives and also consider balancing between robustness and the nominal quality.

For illustrating the computed lightly robust efficient solutions in step 6, we augment the value path visualization [27] to help the decision maker to understand the essence of the solutions. Figure 2 shows an example of visualizing a lightly robust efficient solution with its corresponding nominal objective vector and other related information. In the example figure, there are three objectives represented by bars. The minima and maxima of the bars represent the nominal ideal and nadir values respectively. The filled part of the bar is the objective function value of the nominal efficient solution which satisfies the reference point best. The triangles mark the tolerable degradations (i.e., the value until which the decision maker is willing to sacrifice to gain robustness). The value path illustrates the nominal objective vector of the current lightly robust efficient solution. The gain in robustness (marked as g in the figure) and the price to be paid for robustness (marked as p in the figure) are illustrated by the horizontal bars on the upper left corner.

When a decision maker sees the visualization, (s)he considers the nominal objective function values of the lightly robust efficient solution. This is because of the nature of concentration on the nominal scenario in light robustness. If the values are acceptable, (s)he then further considers if the gained robustness is worthy of the sacrifice in the nominal quality by simply comparing the lengths of the bars marked by g and p. If the decision maker asks, we can present their numerical values.

To summarize, in the figure, the decision maker can see five types of information:

- The nominal objective function values of the current nominal efficient solution which satisfies the reference point best in the colored bars.
- The nominal objective function values of the current lightly robust efficient solution.
- The change in the nominal quality, i.e., the difference between the markers of the value path and the filled part of the bars.
- How much better the current lightly robust efficient solution is compared to the worst acceptable one.
- How much robustness (s)he has gained in the solution compared to the sacrificed nominal quality.

The five types of information together help the decision maker to understand the current lightly robust efficient solution in terms of its nominal quality, the relationships with the nominal efficient solution and the trade-offs between robustness and nominal quality. They also help the decision maker to analyze what kind of changes (s)he should make to get a more desired solution by specifying the preferences for the next iteration. If (s)he sees that the tolerable degradations can be still modified while the gain in robustness is not sufficient, (s)he can relax the value of ε to get a more robust solution. If (s)he is not willing to sacrifice more in the nominal quality or is not satisfied by the nominal objective function values of the lightly robust efficient solution, (s)he can try to provide a new reference point to change the nominal efficient solution.

After presenting the lightly robust efficient solution, if the decision maker is satisfied, we terminate the solution process with $x^{(\text{light},\varepsilon)_c}$ as the final solution. If the decision maker wishes to continue the solution process, we calculate a new solution based on her or his preferences. By interacting with the decision maker, we support her or him to find a satisfactory lightly robust efficient solution based on her or his preferences on the nominal objective function values and the tolerable degradations. If the decision maker so desires, the objective vector of the final solution in the worst case can also be presented. In case that the worst case objective vector is not acceptable, the decision maker can provide new preferences to get a new solution.

5 Example in investment portfolio optimization

Portfolio optimization problems have been considered in the literature but with different concepts of robustness. In this section, we formulate a simple investment portfolio optimization problem with uncertainty in future developments. We solve this problem to demonstrate the ability of the LiRoMo method in supporting the decision maker to find a most preferred balance between robustness and the nominal quality.

The main products of a start-up Company A are software products. Now the owners of the company are considering investing in some stocks for long-term return. Just like in any portfolio investment problem, they want to maximize the return on investment and minimize the risks. They plan to study the long-term historical data of the stocks to find a good composition for their investment portfolio. However, it is also possible that their investment will be withdrawn as a mid-term or even a short-term investment if they discover and initiate a new interesting project. Here, the uncertainty comes from the fact that Company A does not know exactly which of the time frames should be used in the portfolio optimization. In this case, we have three discrete scenarios in the uncertainty set including the short-term data, mid-term data, and long-term data. Company A wants to concentrate on a long-term investment, so using long-term data is considered as the nominal scenario and using short- and mid-terms data are considered as the other two possible scenarios.

In the solution process, we aim at finding a composition of investments, which is good with respect to all scenarios and especially not too bad as a long-term investment. We apply the LiRoMo method and interact with a decision maker to find a lightly robust investment portfolio with good return and acceptable risks in long-term but at the same time with not too bad return and risks at any earlier withdrawal.

As mentioned before, the objectives of the investment include maximizing the return on investment and minimizing the risks. Different risk measures providing different insights are used in the literature such as the standard deviation, β index, Sharpe index, and Treynor index, etc (see e.g., [29]). In this paper, we use the Sharpe index and the Treynor index to be maximized as the standard deviation is related to the Sharpe index and the β index is related to the Treynor index and we only consider indices with a low correlation. Briefly speaking, the Sharpe index indicates how well a portfolio uses risk to get return and the Treynor index measures the volatility in the market to calculate the value of a portfolio adjusted risk. In order to avoid any failure in unexpected behaviors in a single or a few good stocks, we also minimize the maximum amount of investment in a single stock among all the invested stocks. The problem is formulated as a multiobjective optimization problem with uncertain parameters in the objectives:

$$\begin{array}{ll}
\text{maximize} & f_1(x) = \sum_{i=1}^{n} \frac{p_{ij}^{t} - p_{ij}^{t-1}}{p_{ij}^{t-1}} x_i \\
\text{maximize} & f_2(x) = \frac{1}{\sigma_j} \left(\sum_{i=1}^{n} \frac{p_{ij}^{t} - p_{ij}^{t-1}}{p_{ij}^{t-1}} x_i - \tilde{r} \right) \\
\text{maximize} & f_3(x) = \frac{1}{\beta_j} (\bar{r}_j - \tilde{r}) \\
\text{minimize} & f_4(x) = \max_i x_i \\
\text{subject to} & x_i \geq 0 \\
& \sum_{i=1}^{n} x_i = 1 \\
& p_{ij}^{t} \in P_i^{t} \\
& p_{ij}^{t-1} \in P_i^{t-1}.
\end{array} \tag{13}$$

In the formulation, f_1 represents the return on investment, f_2 represents the Sharpe index, f_3 represents the Treynor index, and f_4 represents the maximum investment in a single stock. In this formulation, if the value of f_1 is smaller than 1, there is loss in the investments. The decision variables x_i represents the proportion of the total amount of investment in the stock i and there are in total n stocks for investment. The parameters p_{ij}^{t}, p_{ij}^{t-1} are the historical buying and selling prices of the stock i when the time frame j is used, where t stands for the most recent time and $t-1$ represent the previous time period. The notation $\tilde{r}$ represents the risk free rate and $\bar{r}_j$ is the average return rate of the portfolio when the time frame j is used. The standard deviation of the return on the portfolio in the time frame j is denoted by σ_j. The beta index is denoted by β_j in the time frame j. The value of the beta index depends on the return on investment of the

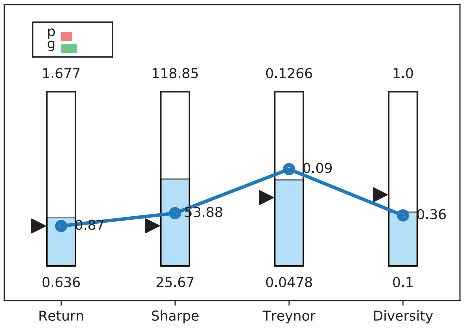


Figure 3: Initial solution

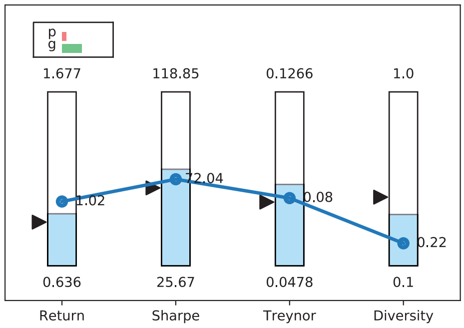


Figure 4: Iteration 1

investment portfolio.

The parameters can have different values depending on the time frame used. Each parameter has three different possible values to consider, i.e., when short-, mid-, and long-term data are used in optimizing the portfolio. So, we have two uncertainty sets P_i^t and P_i^{t-1} and each uncertainty set has three elements. For example, we have $P_i^t = \{p_{is}^t, p_{im}^t, p_{il}^t\}$ where s stands for the short time frame, m stands for the mid-term time frame and l stands for long time frame. The nominal values of the sets are p_{il}^t and p_{il}^{t-1}. The indices σ_j, and β_j depend on the data in the time frame (i.e., p_{ij}^{t-1} and p_{ij}^t). Note that in the formulation, $f_4(x)$ does not involve uncertain parameters but different investment portfolios (i.e., solutions) have correspond to different values of $f_4(x)$. Problem (13) can be reformulated for computing lightly robust efficient solutions based on problem (5).

As for the historical data of the portfolios, we downloaded 10 different NASDAQ stocks from Google finance [1] and computed the needed indices. We started our solution process by calculating the nominal ideal objective vector and approximating the nadir objective vector with the payoff table. We have $z^{\text{ideal}} = (1.677, 118.85, 0.1266, 0.1)^T$ and $z^{\text{nad}} = (0.636, 25.665, 0.0478, 1)^T$. We also calculated and approximated the robust ideal and nadir objective vectors $z^{\text{ideal,wc}} = (1.34, 89.67, 0.093, 0.1)^T$ and $z^{\text{nad,wc}} = (0.38, 23.37, 0.0289, 1)^T$. One should note that since the first three objectives are to be maximized, the corresponding components in the ideal objective vector are higher than those of the nadir objective vector. In this problem, the uncertain parameters stem from discrete sets, so we utilized the reformulation (10) to compute the lightly robust efficient solutions.

Initialization. We presented the nominal ideal and nadir objective vectors to the decision maker as the ranges of the objective function values of the nominal efficient solutions. We also presented the robust ideal and nadir objective vectors to the decision maker as the ranges in the minmax robust efficient solutions.

Initial solution. Based on the ranges, the decision maker specified the reference point: $\bar{z}^0 = (1.1, 75, 0.1, 0.5)^T$. With the preferences, we first solved (8) and found $\hat{x}^0$. Then, we presented the nominal objective function values of $\hat{x}^0$ to the decision maker and asked him to specify his preferences on the tolerable degradation. Then, we solved the reformulated lightly robust problem and found an initial lightly robust efficient solution and presented it to the decision maker as shown in Figure 3. Based on this solution, the decision maker could not accept the loss in the investments even though the price to be paid for robustness resulted in a good gain in robustness.

Iteration 1. So, we asked him to specify a new reference point. Based on the new reference point $\bar{z}^1 = (1.1, 85, 0.1, 0.35)^T$, we calculated $\hat{x}^1$. Based on on the objective function values of $\hat{x}^1$, the decision maker specified the tolerable degradations. With the

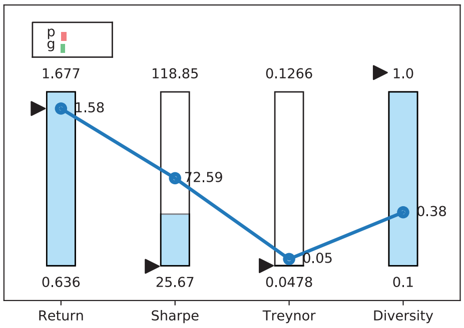


Figure 5: Iteration 2

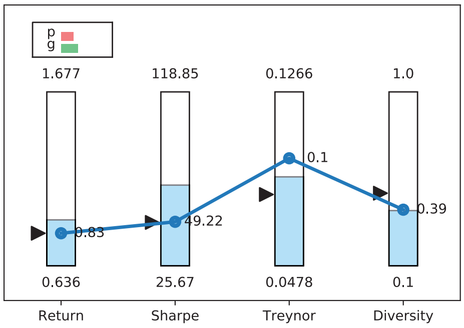


Figure 6: Iteration 3

tolerable degradations $\epsilon^1 = (0.05, 10, 0.008, 0.09)^T$ and $\hat{x}^1$, we calculated a new lightly robust efficient solution by solving (10). We presented its nominal objective vector to the decision maker as in Figure 4. With the solution, the decision maker could make only little profit even though he noticed that the price to be paid for robustness resulted in a much bigger amount of gain in robustness.

Iteration 2. The decision maker decided to try to provide a really good aspiration level to the return on investment with less emphasis on other objectives. With $\bar{z}^2 = (1.8, 20, 0.05, 1)^T$, we calculated $\hat{x}^2$. Based on $f^{\text{nom}}(\hat{x}^2)$, the decision maker specified $\epsilon^2 = (0.1, 28, 0, 0.1)^T$, we again solved (10) and found a new lightly robust efficient solution as illustrated in Figure 5. The decision maker noticed that even though the return on investment was good, the investment was rather risky and the lightly robust efficient solution was very different from the nominal efficient solution. Since the nominal objective function values of the lightly robust efficient solution were not acceptable, he did not even consider the trade-off between robustness and the nominal quality.

Iteration 3. The decision maker provided a new reference point $\bar{z}^3 = (1.2, 40, 0.11, 1)^T$ with which we computed $\hat{x}^3$. With the presented $f^{\text{nom}}(\hat{x}^3)$, the decision maker specified the tolerable degradations $\epsilon^3 = (0.08, 20, 0.008, 0.09)^T$. Based on $\hat{x}^3$ and ϵ^3, we calculated and presented a new lightly robust efficient solution to the decision maker as in Figure 6. In this solution, the decision maker noticed that the Treynor index increased as she intended. However, there were losses in the investment. As before, since this solution was not acceptable, he did not consider comparing the values of gain in robustness and price to be paid for robustness.

Iteration 4. The decision maker tried with another reference point with a better aspiration level on return on investment $\bar{z}^4 = (1.5, 85, 0.1, 0.35)^T$. After knowing $f^{\text{nom}}(\hat{x}^4)$ which we computed, he also decided not to allow the return on investment and the Sharpe index to degrade as much as in previous iterations with $\epsilon^4 = (0.05, 10, 0.008, 0.09)^T$. Using $\hat{x}^4$ and ϵ^4, we found a new lightly robust efficient solution as in Figure 7. The decision maker liked the solution in terms of its nominal objective function values and the balance in the price to be paid for and the gain in robustness.

Termination. Even though the decision maker liked the solution in the previous iteration, he still provided a new reference point $\bar{z}^5 = (1.7, 55, 0.1, 0.25)^T$ to try to increase the return on investment by lowering the Sharpe index. After knowing $f^{\text{nom}}(\hat{x}^5)$, he decided to further reduce the tolerable degradations of the return on investment and the Sharpe index with $\epsilon^5 = (0.04, 5, 0.008, 0.09)^T$. With the new preference information, the solution presented in Figure 8 was obtained. The decision maker found that this lightly robust efficient solution had better nominal objective function values than in the previous iteration with the reduced tolerable degradations. In addition, the gain in robustness was

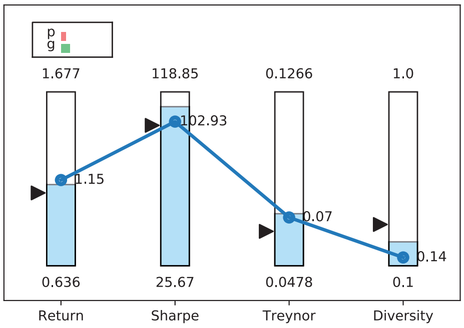


Figure 7: Iteration 4

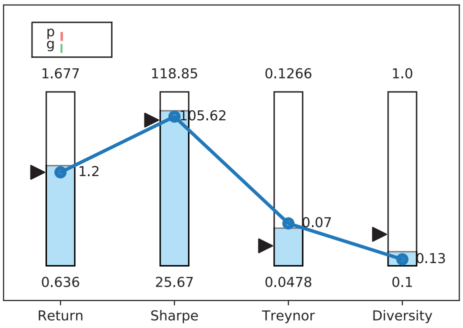


Figure 8: Final solution

worthy of the sacrifice in the nominal quality with a preferred balance. So, he decided to terminate the solution process.

During the solution process, the decision maker chose to provide reference points and tolerable degradations for the computation of lightly robust efficient solutions. In the beginning of the solution process, the decision maker took some iterations to learn about the attainable objective function values and how he could utilize the reference point and the value of ε to guide the solution process toward the kind of solutions she desires. In the last two iterations, the decision maker was able to find a desirable lightly robust efficient solution and utilize ε to fine-tune the solution. In addition, the comparison between the price to be paid for and the gain in robustness served as useful information for the decision maker to make an informed decision. The final solution found was satisfactory with a good return on investment and acceptable risk levels.

6 Conclusions

In this paper, we considered multiobjective optimization problems where some parameters are uncertain in the objective functions. In order to support the decision maker to find a most preferred solution with a good balance between robustness and the nominal quality, we developed the LiRoMo method by utilizing the concept of light robustness. It is the first interactive method using light robustness. In the LiRoMo method, the decision maker can alter the trade-offs between robustness and the nominal quality by changing tolerable degradations in the nominal quality. As a support for the decision maker to consider the balance, we quantified the price to be paid for robustness and the gain in robustness in each computed lightly robust efficient solution. In addition, we visualized the lightly robust efficient solutions and related information to help the decision maker to understand them with an augmented value path visualization.

We formulated an investment portfolio optimization problem and solved it with the LiRoMo method involving a decision maker to demonstrate the advantages of the method. With the support provided by the method, the decision maker was able to explore the objective function values of the solutions of the problem and eventually found a lightly robust efficient solution with a good balance between robustness and the nominal quality.

In this paper, we reformulated the lightly robust problem based on the achievement scalarizing function under some assumptions: the objective functions are quasi-convex with respect to the uncertain parameter with a fixed decision vector and the uncertain parameters stem from polyhedral uncertainty sets. An immediate continuation of this research is to efficiently compute lightly robust solutions for more general problems. In the current version of the interactive method, it is possible that the trade-offs between the objective function values of the lightly robust efficient solutions are different from those of the nominal efficient solution. The reason is that the preferences on the nominal objective function values are first considered in the form of a reference point. When calculating the lightly robust efficient solution, the focus is on the robustness. Thus, another interesting future research direction is to refine interactive methods to further maintain the preferences set in the nominal objective function values in the lightly robust efficient solutions.

Acknowledgments

We thank Dr. Dmitry Podkopaev for various discussions in formulating the investment portfolio optimization problem.

References

[1] Google finance. URL `http://https://www.google.com/finance`.

[2] A. Ben-Tal and A. Nemirovski. Robust convex optimization. *Mathematics of Operations Research*, 23(4):769–805, 1998. doi: 10.1287/moor.23.4.769.

[3] A. Ben-Tal, A. Goryashko, E. Guslitzer, and A. Nemirovski. Adjustable robust solutions of uncertain linear programs. *Mathematical Programming*, 99(2):351–376, 2004. doi: 10.1007/s10107-003-0454-y.

[4] A. Ben-Tal, L. El Ghaoui, and A. Nemirovski. *Robust Optimization*. Princeton University Press, 2009.

[5] A. Ben-Tal, D. Bertsimas, and D. B. Brown. A soft robust model for optimization under ambiguity. *Operations Research*, 58(4-part-2):1220–1234, 2010. doi: 10.1287/opre.1100.0821.

[6] A. Ben-Tal, E. Hazan, T. Koren, and S. Mannor. Oracle-based robust optimization via online learning. *Operations Research*, 63(3):628–638, 2015. doi: 10.1287/opre.2015.1374.

[7] D. Bertsimas and M. Sim. The price of robustness. *Operations Research*, 52(1):35–53, 2004. doi: 10.1287/opre.1030.0065.

[8] G. R. Bitran. Linear multiple objective problems with interval coefficients. *Management Science*, 26(7):694–706, 1980. doi: 10.1287/mnsc.26.7.694.

[9] R. Bokrantz and A. Fredriksson. Necessary and sufficient conditions for Pareto efficiency in robust multiobjective optimization. *European Journal of Operational Research*, 262(2):682–692, 2017. doi: 10.1016/j.ejor.2017.04.012.

[10] J. Branke, K. Deb, K. Miettinen, and R. Slowinski, editors. *Multiobjective Optimization, Interactive and Evolutionary Approaches*, 2008. Springer.

[11] K. Deb, K. Miettinen, and S. Chaudhuri. Toward an estimation of nadir objective vector using a hybrid of evolutionary and local search approaches. *IEEE Transactions on Evolutionary Computation*, 14(6):821–841, 2010. doi: 10.1109/TEVC.2010.2041667.

[12] G. M. Dranichak and M. M. Wiecek. On highly robust efficient solutions to uncertain multiobjective linear programs. *European Journal of Operational Research*, 2018. doi: https://doi.org/10.1016/j.ejor.2018.07.035.

[13] M. Ehrgott, J. Ide, and A. Schöbel. Minmax robustness for multi-objective optimization problems. *European Journal of Operational Research*, 239(1):17–31, 2014. doi: 10.1016/j.ejor.2014.03.013.

[14] G. Eichfelder, C. Krüger, and A. Schöbel. Decision uncertainty in multiobjective optimization. *Journal of Global Optimization*, 69(2):485–510, 2017. doi: 10.1007/s10898-017-0518-9.

[15] M. Fischetti and M. Monaci. Light robustness. In R. K. Ahuja, R. H. Möhring, and C. D. Zaroliagis, editors, *Robust and Online Large-Scale Optimization: Models and Techniques for Transportation Systems*, pages 61–84. Springer, 2009. doi: 10.1007/978-3-642-05465-5_3.

[16] J. Fliege and R. Werner. Robust multiobjective optimization & applications in portfolio optimization. *European Journal of Operational Research*, 234(2):422–433, 2014. doi: j.ejor.2013.10.028.

[17] M. Goberna, V. Jeyakumar, G. Li, and J. Vicente-Pérez. Guaranteeing highly robust weakly efficient solutions for uncertain multi-objective convex programs. *European Journal of Operational Research*, 270(1):40 – 50, 2018. doi: https://doi.org/10.1016/j.ejor.2018.03.018.

[18] M. Goerigk and A. Schöbel. Algorithm engineering in robust optimization. In L. Kliemann and P. Sanders, editors, *Algorithm Engineering: Selected Results and Surveys*, pages 245–279. Springer, 2016. doi: 10.1007/978-3-319-49487-6_8.

[19] J. Granat, K. Kreglewski, J. Paczynski, and A. Stachurski. IAC-DIDASN++ modular modeling and optimization systems theoretical foundations. *Report of the Institute of Automatic Control, Warsaw University of Technology*, 1994.

[20] W. J. Gutjahr and A. Pichler. Stochastic multi-objective optimization: a survey on non-scalarizing methods. *Annals of Operations Research*, 236(2):475–499, 2016. doi: 10.1007/s10479-013-1369-5.

[21] F. Hassanzadeh, H. Nemati, and M. Sun. 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. doi: 10.1016/j.ejor.2014.03.023.

[22] J. Ide and A. Schöbel. Robustness for uncertain multi-objective optimization: a survey and analysis of different concepts. *OR Spectrum*, 38(1):235–271, 2016. doi: 10.1007/s00291-015-0418-7.

[23] M. Inuiguchi, H.Ichihashi, and H. Tanaka. Fuzzy programming: A survey of recent developments. In H. Shi-Yu and T. Jaques, editors, *Stochastic Versus Fuzzy Approaches to Multiobjective Mathematical Programming under Uncertainty*, pages 45–68. Springer, 1990. doi: 10.1007/978-94-009-2111-5_4.

[24] K. Kuhn, A. Raith, M. Schmidt, and A. Schöbel. Bi-objective robust optimisation. *European Journal of Operational Research*, 252(2):418–431, 2016. doi: 10.1016/j.ejor.2016.01.015.

[25] D. Kuroiwa and G. M. Lee. On robust multiobjective optimization. *Vietnam Journal of Mathematics*, 40(2-3):305–317, 2012.

[26] K. Miettinen. *Nonlinear Multiobjective Optimization*. Kluwer Academic Publishers, 1999.

[27] K. Miettinen. Survey of methods to visualize alternatives in multiple criteria decision making problems. *OR Spectrum*, 36(1):3–37, 2014. doi: 10.1007/s00291-012-0297-0.

[28] H. Nakayama and Y. Sawaragi. Satisficing trade-off method for multiobjective programming and its applications. *IFAC Proceedings Volumes*, 17(2):1345–1350, 1984. doi: 10.1016/S1474-6670(17)61162-5.

[29] F. K. Reilly and K. C. Brown. *Investment Analysis and Portfolio Management*. Cengage Learning, 2011.

[30] A. Schöbel. Generalized light robustness and the trade-off between robustness and nominal quality. *Mathematical Methods of Operations Research*, 80(2):161–191, 2014. doi: 10.1007/s00186-014-0474-9.

[31] T. W. Simpson, D. K. Lin, and W. Chen. Sampling strategies for computer experiments: Design and analysis. *International Journal of Reliability and Applications*, 2 (3):209–240, 2001.

[32] A. L. Soyster. Technical note - Convex programming with set-inclusive constraints and applications to inexact linear programming. *Operations Research*, 21(5):1154–1157, 1973. doi: 10.1287/opre.21.5.1154.

[33] R. E. Steuer. *Multiple Criteria Optimization: Theory, Computation, and Applications*. John Wiley & Sons, Inc., 1986.

[34] R. E. Steuer and E.-U. Choo. An interactive weighted Tchebycheff procedure for multiple objective programming. *Mathematical Programming*, 26(3):326–344, 1983. doi: 10.1007/BF02591870.

[35] H. Tuy. *Convex Analysis and Global Optimization*. Springer, 2016.

[36] M. M. Wiecek and G. M. Dranichak. Robust multiobjective optimization for decision making under uncertainty and conflict. In A. Gupta, A. Capponi, J. C. Smith, and H. J. Greenberg, editors, *Optimization Challenges in Complex, Networked and Risky Systems*, pages 84–114. INFORMS, 2016. doi: 10.1287/educ.2016.0153.

[37] A. P. Wierzbicki. On the completeness and constructiveness of parametric characterizations to vector optimization problems. *OR Spectrum*, 8(2):73–87, 1986. doi: 10.1007/BF01719738.

[38] Y. Zhou-Kangas and A. Schöbel. The price of multiobjective robustness: Analyzing solution sets to uncertain multiobjective problems. Manuscript, 2018.